

# DRAFT SF 298

1. Report Date (dd-mm-yy) October 1995		2. Report Type		3. Dates covered (from... to )	
4. Title & subtitle Requirements Engineering and Design Technology Report				5a. Contract or Grant #	
				5b. Program Element #	
6. Author(s) Ganska, R; Grotzky, J; Rubinstein, J; Van Buren, J				5c. Project #	
				5d. Task #	
				5e. Work Unit #	
7. Performing Organization Name & Address				8. Performing Organization Report #	
9. Sponsoring/Monitoring Agency Name & Address Software Technology Support Center OO-ALC/TISE 7278 4th Street Hill AFB, UT 84056-5205				10. Monitor Acronym	
				11. Monitor Report #	
12. Distribution/Availability Statement Distribution Statement A: Approved for public release, distribution is unlimited.					
13. Supplementary Notes					
14. Abstract					
15. Subject Terms					
Security Classification of			19. Limitation of Abstract	20. # of Pages	21. Responsible Person (Name and Telephone #)  Randy Wright (801) 777-9732
16. Report Unclass	17. Abstract Unclass	18. This Page Unclass			

19970516 095

STSC

---

Requirements Engineering and  
Design Technology Report

October 1995

---

Ralph Ganska  
John Grotzky  
Jack Rubinstein  
Jim Van Buren

*Upper CASE Products Evaluation Project*

Gary Petersen  
*Technical Program Manager*

19970516 095

This technical report was prepared by the Software Technology Support Center at the following address:

Software Technology Support Center  
Ogden ALC/TISE  
7278 4th Street  
Hill AFB, UT 84056-5205

The ideas and findings in this report should not be construed as an official Air Force position. It is published in the interest of scientific and technical information exchange.

This document is available through the STSC. To obtain a copy, please contact the STSC Customer Service Office directly at: Software Technology Support Center, Attn: Customer Service, Hill AFB, Utah 84056; 801-775-5555 x3039 or DSN 775-5555, Fax 801-777-8069, E-mail [redteam@software.hill.af.mil](mailto:redteam@software.hill.af.mil).

## Preface

This document, Requirements Engineering and Design Technology Report 1995, is a synopsis of the progress of the Software Technology Support Center (STSC) in evaluating requirements analysis and preliminary design computer-aided software engineering products. Throughout this report we will refer to these requirements analysis and preliminary design computer-aided software engineering products as Upper CASE products. The targets of this report are organizations responsible for the development and maintenance of computer software. This report defines the Upper CASE products and identifies their value in improving software quality. It explains how the features of current Upper CASE products can improve software development and maintenance. It includes information about specific products in the marketplace. The information is aimed at those who must make the decisions about acquiring advanced technology and prepare their organizations for its effective use. Finally, this report attempts to identify the future directions of the field to help plan long-range strategies.

The content of this document has changed little from its predecessor, Requirements Analysis and Design Tools Report 1994. The product lists and product information sheets have been completely updated or reverified. There has been some reorganization of the material to reflect growing standardization of STSC technology reports.

Although the material presented in this publication has been reviewed for technical accuracy, no guarantees are made or implied. Product specifications are subject to change by the vendor without notice. Readers should independently verify this information and evaluate it in relationship to their environment.

---

# Contents

---

Preface .....	iii
<b>1 Requirements Analysis and High Level Design Software Engineering Technology .....</b>	<b>1</b>
<b>1.1 Upper CASE Product Classification.....</b>	<b>1</b>
1.1.1 Lifecycle Phase .....	2
1.1.2 Methodology .....	3
1.1.3 Application Domain .....	3
1.1.4 Functional Capabilities and Product Quality.....	5
<b>2 Upper CASE Products .....</b>	<b>6</b>
2.1 Upper CASE Products – Product List.....	6
2.2 Upper CASE Products – Product Sheets.....	6
2.3 Upper CASE Products – Product Critiques.....	7
<b>3 Selection and Use of Upper CASE Products.....</b>	<b>8</b>
3.1 When to Use Upper CASE Products .....	8
3.2 How to Select Upper CASE Technology.....	9
3.3 How to Use Upper CASE Products .....	11
<b>4 Future Directions.....</b>	<b>13</b>
<b>5 Conclusions.....</b>	<b>15</b>
<b>Appendix A: Upper CASE Products – Product List .....</b>	<b>17</b>
<b>Appendix B: Upper CASE Product Sheets .....</b>	<b>27</b>
<b>Appendix C: Upper CASE Products – Product Critique Format</b>	<b>113</b>
<b>Appendix D: Upper CASE Product Characteristics.....</b>	<b>117</b>
<b>Appendix E: Recommended Readings .....</b>	<b>129</b>
<b>Appendix F: Glossary of Terms .....</b>	<b>135</b>

---

## Contents (continued)

<b>Appendix G: Software Technology Support Center .....</b>	<b>143</b>
<b>G.1 Software Technology Support Center (STSC) Overview.....</b>	<b>144</b>
G.1.1 The Software Technology Support Center.....	144
<b>G.1.2STSC Technology Transition Approach.....</b>	<b>146</b>
G.1.2.1 Technology Evaluation.....	146
G.1.2.2 Information Exchange.....	146
G.1.2.2.1 CrossTalk.....	146
G.1.2.2.2 Software Technology Conference .....	147
G.1.2.2.3 Technology Reports.....	147
G.1.2.2.4 Electronic Customer Services.....	147
G.1.2.3 Technology Insertion Projects.....	149
G.1.2.4 STSC Associates.....	150
<b>G.1.3 Embedded Computer Resources Support Improvement Program</b>	
<b>(ESIP).....</b>	<b>151</b>

(This page is intentionally left blank.)

# **1 Requirements Analysis and High Level Design Software Engineering Technology**

This report reviews the STSC's recommendations for the selection and usage of software engineering products aimed at the requirements analysis and high-level design portions of the software lifecycle. In the early and mid-1980s, when these products first became available, they were called CASE (Computer-Aided (or Assisted or Automated) Software (or Systems) Engineering) products because they were the first class of products to automate software engineering practices. Later the term Upper CASE was coined to differentiate these products from those that targeted the later phases of the waterfall software lifecycle model. Our use of the term Upper CASE should not be construed as an endorsement of the waterfall lifecycle model. It is used only to represent the aggregation of requirements analysis and high-level design products.

This report focuses on two primary issues. The first, how to select and use Upper CASE products, is intended to provide general guidance to the reader. It describes a process for selecting products and contains anecdotal examples of their use. The examples provide analysis of why Upper CASE product insertions have failed in the past. The second focus of the report is an enumeration and classification of Upper CASE products. The report does not address the very important and often overlooked managerial aspects of methodology selection and training. The report does not justify the use of Upper CASE products either through anecdotal examples or through empirical analysis. Appendix E contains a reading list that does address these issues.

## **1.1 Upper CASE Product Classification**

Upper CASE products can be classified in a number of useful ways. The STSC uses five categories of attributes and features to differentiate and select Upper CASE products. They are:

- Lifecycle Phase
- General Methodology
- Intended Application
- Functional Capabilities
- Product Quality



While these attribute/feature categories are designed to be independent from each other, there are some interrelationships. The first three types – lifecycle phase, general methodology, and intended application – capture information about the basic assumptions of the product developers. These are discussed in detail in Sections 1.1.1, 1.1.2, and 1.1.3. The latter two types, functional capabilities and product quality, in contrast to dealing with the intentions of the product developer, deal with what the product actually does and how well it does it. Functional capabilities and product quality are discussed in Section 1.1.4.

There are many other types of useful product classifications. Examples are types of hardware platforms, types of software platforms, and interoperability with specific other software engineering products or environments. The STSC has not delineated all possible classifications or collected product specific information in any of these "other" classifications. However, we have collected information regarding the hardware and software platforms necessary to support a product.

### **1.1.1 Lifecycle Phase**

Upper CASE products, by our definition, are products that address activities specific to the requirements analysis and high-level design lifecycle phases. The STSC considers identifying the lifecycle phase that a product supports and classifying products by the supported lifecycle phase as a useful aid in making a product selection. While the requirements analysis and high-level design lifecycle phases are generally associated with the waterfall software lifecycle model, we define them here in a nonspecific lifecycle model fashion. All lifecycle models generally have four active steps: (1) determining what will be built, (2) determining how to build it, (3) building it, and (4) maintaining it. These steps are generally called analysis, design, implementation, and maintenance. Maintenance involves repeating the analysis, design, and implementation steps, but the issues of supporting an existing product necessitate it being identified as a separate step. Testing is identified as an additional step in some lifecycle models. In our lifecycle model, testing is the exit activity for each lifecycle step.

Requirements analysis is the lifecycle step in which a determination is made of what will be built. Issues of software functional capability, as well as compatibility, with other parts of the system are addressed. The testing subactivity determines that a software system can be built that meets the requirements, i.e., that the requirements are complete and consistent, and that the requirements are testable in later lifecycle phases. Design is the portion of the lifecycle in which a specification of how the system will be built is created. The design phase is usually

broken into two steps: high-level (also sometimes called preliminary or architecture design) and low-level design (also sometimes called detail design). In high-level design, general algorithms (algorithms contain all three major design considerations: control, data, and state) are developed. These algorithms do not contain implementation details. During low-level design, implementation details (such as target machine architecture timing considerations) are added to the design. Each of these design activities concludes with a testing subactivity that ensures that the design meets its requirements.

### 1.1.2 Methodology

There are three important development views of software systems – object-oriented (data-oriented), process-oriented (functional or structured), and behavior-oriented (temporal, state-oriented, or dynamic). Each of these views takes a different perspective of the system being developed. This supports the notion of appropriately using differing methodology paradigms that support each of the differing views to address a system's development. Development methodologies typically concentrate on one of the paradigms with support for the other two. The selection of a development methodology should be driven by the problem that is being addressed. The selected methodology should be the one that best supports the view for which the problem is most clearly stated. Some products primarily support one methodology viewpoint, while others may support two or all three.

### 1.1.3 Application Domain

Products are often appropriate to specific application domains. Types of application domains are:

- Embedded
- Communications, Command, Control, and Intelligence (C<sup>3</sup>I)
- Commercial Embedded
- Scientific/Engineering/Technical
- Parallel or Distributed
- Artificial Intelligence (Expert Systems)
- Commercial (MIS, Transaction Processing)

The Embedded domain is characterized by applications that *require* response to a stimulus within a specified time period. Example applications include avionics control, radar, and military fire control. These applications are typified by requirements that emphasize timing

issues and extreme design and implementation timing concerns. As software timing requirements begin to approach theoretical hardware speeds, the analysis, design, and testing of timing concerns become critical to the successful implementation of these applications. The military has traditionally called these types of applications "Embedded." We use this terminology. Academic and other nonmilitary organizations sometimes use the terminology "Hard Real Time" or "Real Time/Reactive" to refer to this domain.

The C<sup>3</sup>I applications domain is much like the Embedded domain in that the software is part of a larger system and there may be timing constraints. It differs in that in C<sup>3</sup>I applications much more of the system functionality is implemented by software. Example applications include air tasking order generation, mission planning, satellite image interpretation, and battlefield communications.

Applications in the Commercial Embedded domain are similar to embedded applications. Both domains share the attribute of software being a part of a larger physical system. They differ in that Commercial Embedded applications do not have the severe timing constraints of Embedded applications. Consumer products such as videocassette recorders, modern washing machines, and microwave ovens are examples of embedded applications. Outside of the military, this domain is sometimes called (just) "Embedded" or "Real Time."

The Scientific/Engineering/Technical domain is characterized by applications that support engineers and scientists in the creation of products or knowledge. Computer-Aided Design (CAD), Computer-Aided Manufacturing (CAM), and data analysis and plotting packages are examples of technology that support this domain.

The Parallel (or Distributed) application domain is characterized by applications involving multiple processors including client/server applications, an area of rapid growth. The interactions between the processors are often the most important analysis/design/implementation concern.

The Artificial Intelligence (including Expert Systems) domain generally addresses problems where no algorithmic solutions exist. These problems may have very large solution spaces, where the specialized search techniques of artificial intelligence are appropriate.

The Commercial application domain includes all applications that deal with business data. MIS, Decision Support Systems, and Transaction Processing applications are included in

this domain. These applications handle the receipt, extraction, storage and analysis of data, each category having their own focus and approach to data manipulation.

#### **1.1.4 Functional Capabilities and Product Quality**

While the above classifications (lifecycle phase, methodology, and application domain) provide a view of a product's intent, it is still necessary to determine exactly what a product does and how well it does it. To support this concept, a detailed functional taxonomy and a general methodology to evaluate the quality of a product are needed. Appendix D, Upper CASE Product Characteristics, contains an Upper CASE product functional taxonomy and a quality evaluation system.

## **2 Upper CASE Products**

### **2.1 Upper CASE Products – Product List**

An important and necessary step in the technology selection process is to identify candidate products. The current list of 144 Upper CASE products is found in Appendix A, Upper CASE Products – Product List. It includes the product name, vendor, vendor phone number, and some general information about the product. The general information identifies the product's intended use, its intended audience, and what type of platforms it runs on. Due to the very dynamic nature of the CASE product business, the list will contain inaccuracies and omissions at its publication. If you are aware of any, please contact the STSC's Upper CASE team. They can also be contacted for up-to-date lists.

The list was developed from three types of sources: personal experiences, product literature, and conferences. The initial sources for the list were the personal notes, experiences, and contacts of the project engineers. Industry surveys in periodicals such as *Digital Review*, *Byte*, and *Software News* yielded Upper CASE products outside the application domain (real-time software) of the project engineers. STSC members attended several conferences to identify the products that are actively being marketed. Many other conference announcements were studied to identify additional product vendors.

The information in the list was distilled from the product's product sheet (PS). The data collection methodologies used in building the PSs are discussed in Section 2.2. The general information included in the list identifies a product's intended use, its intended audience, and what type of platforms it runs on.

### **2.2 Upper CASE Products – Product Sheets**

Appendix B, Upper CASE Product Sheets, contains the PSs for most of the Upper CASE products in the product list. These reports provide more detailed information than the product list. Users of these reports can make preliminary product assessments based on the information provided. Information on pricing, contacts, support, addressed lifecycle phases and activities, intended audiences, methodologies, hardware platforms, and general product capabilities is included. The information in the reports was obtained either directly from the vendor or from the vendor's literature. In some cases, the vendor has authenticated the

information. There are products in the product list for which there is no associated product sheet. This condition occurs because there was no available information to create the product sheet either because the vendor did not supply any or because the product was added to the product list too late for the creation of a product sheet.

The STSC can be contacted for both unpublished and updated reports that may be available.

## **2.3 Upper CASE Products – Product Critiques**

The STSC is soliciting and using product critiques from experienced product users. These critiques highlight the experiences (both good and bad) of actual product users. If you are a user of a product that is or should be in the product list and would like to write a critique, please contact the STSC. An example product critique is found in Appendix C, Upper CASE Product Critique Format. The Upper CASE team has collected 12 Upper CASE technology critiques. The tools that have been critiqued are listed in Appendix C. The Upper CASE team should be contacted for these and additional critiques.

## 3 Selection and Use of Upper CASE Products

### 3.1 When to Use Upper CASE Products

Upper CASE products are an appropriate technology when three conditions have been satisfied. These conditions are (1) a need for the software requirements analysis and design processes to be performed, (2) an ability to automate these processes, and (3) a need to automate these processes. It should be emphasized that having a well-defined software development process will greatly increase the probability of successful technology use. Having no process will almost guarantee failure.

Software requirements analysis and design are necessary (required for military software) for all nontrivial long-lived software systems. Requirements analysis produces a documented description of *what* the software system is supposed to do. It occurs before design and implementation activities. It acts as a contract between the software's consumer and producer. The feasibility of building the software is determined during this process. The outputs of requirements analysis are used by the software designers to produce a software design and (theoretically) by the software maintainers to understand the basic requirements of the software system.

Software design produces a documented description of *how* a software system will fulfill its requirements. It occurs after requirements analysis and before implementation in the waterfall model. There are two recognized design abstraction levels: high-level design and low-level design. They differ in that high-level design captures implementation independent decisions and low-level design captures implementation dependent details. The design process maps the *what* of the requirements analysis process to a description of *how* that can be used in the implementation process. The feasibility of the design may also be determined during this process.

Given that software requirements analysis or design is necessary, the next issue in considering the applicability of Upper CASE technology is an ability to automate these analysis and design processes. Upper CASE technology in its simplest form is just a mechanism to automate existing processes. An organization must have an existing requirements analysis or design process before considering Upper CASE technology. This is a very important consideration. Historically, many organizations have failed to transition to Upper CASE

technology because they did not have an existing process. They attempted to use the Upper CASE technology to automate their process and to define their process. There are very few successful examples of this approach and many examples of failure.

Finally, there must be a need for the automation of the software requirements analysis or design processes to justify the use of Upper CASE technology. These technologies have an associated cost, which includes far more than the actual dollar cost of the technology. Training costs must be considered and are also easy to quantify. Costs that are not so easy to quantify include, but are not limited to, the complexity of the technology (the engineer must now, in addition to understanding the application or the implementation domain, learn the complexities of the product), the cost of tailoring the technology to the organizations processes, and the risks associated with the technology. The technologies great promise is that they deliver more benefits than their costs. Benefits include, but are not limited to, higher quality analysis and design and greater efficiency in producing and maintaining analyses and designs. There are also nonquantifiable benefits such as greater organizational morale, which is possible to achieve with a well-planned technology insertion sheets.

### **3.2 How to Select Upper CASE Technology**

The STSC's selection of it's Upper CASE products: Product List, Product Sheet, and Product Critique is not accidental. They dovetail into a conceptually simple technology selection strategy that closely mirrors that which would be used by a consumer in conjunction with "Consumer Reports" magazine. It is called the "Consumer Report" selection paradigm and is based on six steps. They are:

- 1) Identify candidate product list
- 2) Identify product requirements
- 3) Shorten candidate product list
- 4) Interview product users
- 5) In-house product test
- 6) Decision



There is actually a step zero, where a domain is identified. In this case the domain is Upper CASE products. This domain was identified through a survey of STSC customers during the first STSC conference in 1989 as a technology domain where the customers needed guidance. In the first step, candidate products are identified. The product list satisfies this criterion. See Appendix D for detail CASE tool selection criteria.

The second step is critical to proper technology selection. In this step the "consumer" identifies the requirements that the technology must satisfy. In the software engineering domain, there are two types of inputs to this requirements specification step: process (or organizational) specific and application specific. The selected technology must satisfy the two broad goals that these inputs address. The technology must fit within the organization's software engineering process plan, so that it will be viable in the long term. It also must be usable on the organization's current applications. As an example of this breakout, in the Upper CASE product domain, consider an organization that must use a particular type of hardware platform for a specific development project and has identified a particular analysis/design methodology (such as Object-Oriented Design or Information Engineering) as an organizational standard for its application domain. The technology requirements are that it support the analysis/design methodology (a process requirement) on the particular hardware platform (an application requirement).

The STSC supplies data to support steps three and four, Shorten candidate product list and Interview product users, in the form of the product reports and the user critiques. The PSs contain sufficient data that the product list can be shortened to include only a small subset of technologies that may be suitable to an organization. The user critiques can be used to gather the adhoc experiences of actual users of the technologies on the shortened list. All this information does not negate the need for in-house demonstrations and test projects of the technology. After the in-house testing, buy decisions can be made.

For Upper CASE products, two product domain characteristics drive the selection process. One characteristic involves general product capability and project product requirements. Products are very niche oriented, where they support only a small subset of the Upper CASE product functionality. Since a project's product requirements are very specific, only a small subset of Upper CASE product characteristics is needed for any project. For example, a project may require methodology "X" and currently has platform "Y" and only one product may support methodology "X" on platform "Y". Thus, there is no opportunity for

quality or cost comparisons. From the product selector's viewpoint, finding a product with the necessary functionality for a specific project often constitutes the product selection process.

The second domain characteristic driving the evaluation guidance process involves the type of work that is done during requirements analysis and high-level design. The work involves capturing information and performing analysis on that information. The quality characteristics that are emphasized are the usability of the product and the correctness of the information capture and analysis functions. Since these are engineering support products, the performance of the product, i.e., speed of execution and resource (disk, memory) usage is not as critical as if the product was used in a production environment.

The Institute of Electrical and Electronics Engineers (IEEE) is developing a recommended practice for the adoption of CASE technology. Parties interested in developing this standard are referred to the chairman of working group P1348, Tom Vollman (301) 862-0798. The standard is expected to be ratified in 1995.

### **3.3 How to Use Upper CASE Products**

Once the need for Upper CASE technology has been demonstrated and a product selected and purchased, then it is time to use the product. The issues of training and technology transition are of particular concern when Upper CASE products are initially used. Upper CASE products and methodologies are very complex. Management must budget and plan for the necessary training, or the product will fail. The old practice of putting a software product on good engineers' desks and letting them learn to use it does not work with Upper CASE products. Proper transition to the new technology is also important. The initial project on which the technology is used must be selected with care. Because of the risk associated with new technology, it should not be used or used with extreme caution on an application where delivery time is critical to the organization. However, the application should be important, so that management cognizance is ensured.

In 1992, the IEEE ratified STD-1209 "Recommended Practice for the Evaluation and Selection of CASE Tools." It is not limited to Upper CASE tools or DoD business practices. The authors of this report helped develop this standard and it is consistent with this report. These IEEE "practice" standards tend to reflect a consensus of the best practices of the day. The standards are valid for five years after which they must be reballoted (after possible

revision). This standard is recommended to readers interested in a more detailed analysis of CASE tool evaluation selection. However, the standard should be critically examined because it only reflects the best available, least controversial knowledge of the 1989-1992 time frame.

## **4 Future Directions**

The Upper CASE technology domain is far from mature. We believe that the root cause of this is the overall immaturity of the software engineering discipline. There is no established cohesive theoretical understanding of how best to develop software. Simply put, products cannot automate what humans do not yet know how to do! The good news is that software engineering is receiving a significant amount of research interest and that a solution or at least an understanding of its problems will occur. The bad news is that only evolutionary and not revolutionary (orders of magnitude) improvements are expected.

To date, the technology of the domain can best be characterized as point solutions to specific problems. This approach proved to be inadequate. Specifically, Upper CASE technology cannot be seamlessly integrated into an organization's software development-maintenance process. The organization must alter its process to use the product, rather than altering the product to fit its process. There are at least four technical areas that the technology must address before it can be seamlessly integrated into a generic organization's software process. They are software process automation, tailorable product standards, product interoperability, and product/process visibility. All of these issues are being addressed by the next generation of Upper CASE products and by other supporting technologies.

The concept of software process as the central theme to improve the production of software products is a relatively new one. It has come into vogue over the last five years. Before this, the central theme was the heavy reliance on analysis and design before implementation to improve the quality of software. The first products to support this theme, the initial generation of Upper CASE products, appeared in the late 1970s and early 1980s. Not unexpectedly, Upper CASE technology still does not support the notion of software process. There are two primary technical shortcomings: (1) the technologies (called software engineering environments or software engineering frameworks) for automating software process are very immature and (2) Upper CASE products are not tailorable to an organization's process or methodologies. A basic paradigm shift must occur within the Upper CASE product domain to address these shortcomings. Currently, the products provide either a de facto software process or assume and support none at all. Upper CASE products must adopt a philosophy that they are a subprocess of the larger software process as defined by the process services of the environment and adopt the interface standards of the environment. They must also continue a trend already begun, where an ability exists to tailor the product's implied process and methodologies to an organization's requirements.

The remaining technical issues facing the next generation of Upper CASE products, tailorable product standards, product interoperability, and product/process visibility, are restatements of current problem areas that will be addressed by the process support as already discussed. The lack of tailorable product standards is a differing viewpoint of the issue of lack of process tailorability. In the latter, there is a rigid format of the output of the product or the types of analysis that may be performed. In the former, there is a rigid process and methodology for producing output or performing analysis. Product interoperability is currently implemented on an adhoc, product-by-product basis. The existence of an environment or framework will provide an interface standard currently lacking. The issue of product/process visibility refers to the lack of current product capabilities to provide an overall picture of the state of a project. Because of its knowledge of the overall process, this function will be provided by an environment or framework.

## **5 Conclusions**

This report summarizes the STSC's work in the area of Upper CASE technologies. A list of Upper CASE products has been developed. A list of product characteristics has been developed, and a framework for evaluating those characteristics has been built. The vendors of the Upper CASE products have provided self-evaluations of their products, which have been modified to reflect STSC and STSC customer experiences.

The STSC's plans for the Upper CASE technology project are threefold: publication of the Upper CASE technology domain report (this report); education of our sponsoring community, Air Force software development/maintenance organizations, on Upper CASE technology concepts; and assistance to software development and support activities in Upper CASE technology selection and adoption. In the past we developed the Upper CASE product domain report. We are now and will continue to maintain that report. This maintenance will reflect the continued evolution and an increased understanding of the product domain. The STSC will continue to update the Upper CASE product list and the product characteristic list. Additional product reports and user critiques will be solicited and published. Emphasis will be placed on methodologies and technologies that have been successfully used on specific applications. Requirements management is one area specifically identified for future research.

Educationally, we will develop and present briefings on Upper CASE technology concepts to interested organizations. Papers that summarize Upper CASE technology will be solicited and developed for general dissemination or presentation at appropriate conferences. Finally, Software Development and Support Activities (SDSAs) will be supported in their product selection and adoption efforts on request.

This report will be republished next year.

(This page is intentionally left blank.)

## **Appendix A: Upper CASE Products – Product List**



This appendix contains a list of Upper CASE products. This list is called the product list. The products are listed alphabetically by vendor name. The list contains general information about the product and the product's vendor. The vendor's name and telephone number are included. Additional information can be obtained in the product's product sheet in Appendix B, Upper CASE Product Sheets. There is also some general information about the product in the list, which identifies the intended use of the product, its intended audience, and on what type of platforms it runs.

Most products are targeted at activities within specific phases of the software lifecycle or activities that occur across lifecycle phases. We have identified four generic software lifecycle activities common to all software lifecycles: Analysis, Design, Coding, and Maintenance. They are not intended to imply any specific lifecycle model such as waterfall or spiral. Reengineering and reverse engineering products are considered maintenance products. We have also identified several activities (Testing, Configuration Management (CM), Documentation, and Project Management) that occur across lifecycle phases. This list is not meant to be inclusive. Other activities may be listed as "Other." Finally, we identified one technology area (Environments or Frameworks) that acts to connect products that automate the various lifecycle phases and activities. A list of abbreviations that occur in the "Type" column appears below. Products may be applicable in one or more activities, so a product may have multiple "Types." More detailed information can be found in the product's PS in Appendix B.

- A: Requirements Specification and Analysis
- D: Design
- C: Coding
- M: Maintenance
- T: Testing
- E: Environment (or Framework)
- U: Documentation and Management Utilities such as CM or Project Management
- O: Other

Vendors usually develop, market, and optimize their products to target specific customer profiles. The "Target" column captures this information. We have identified two primary profiles for the readers of this report, MIS and Technical (TECH). There are two

identified subprofiles for the technical profile: Real Time (RT) and Hard Real Time (HRT). There are many more possible target profiles, such as scientific development, transaction processing, embedded system development, etc., which are collectively identified as "Other" in this list. Some vendors specifically target their products to all market segments. The target for these products is identified as "All." More detailed information can be found in the product's product sheet in Appendix B.

The last general product classification found in the list identifies the type of platform on which the product runs. Three classes of platform have been identified: desktops (DT), workstations (WS), and mainframes (MF). This classification scheme captures the environmental style of the product. Mini-computers are considered workstations in this list. The specific brand of platform is not listed.

Tool	Vendor	Contact Info	Type	Target	Platforms
ADADL	Software Systems Design, Inc.	Tom Radi 909-625-6147	A,D,C,U	All	All
Argos	Versant Object Technology	Andy Smith Tel: 800-837-7268 Fax: 415-325-2380 info@versant.com	A, D, C, E	MIS	WS, DT
ARIS/DesignGen	Software Systems Design, Inc.	Tom Radi 909-625-6147	A,D	All	WS
Axiom-SA	STG, Inc	Vince Peterson Tel: 800-959-2451 Fax: 805-296-5302 info@stgcase.com	A,U	MIS, TECH, RT	DT
AxiomDsn	STG, Inc	Vince Peterson Tel: 800-959-2451 Fax: 805-296-5302 info@stgcase.com	D,U	MIS, TECH, RT	DT
AxiomSys	STG, Inc	Vince Peterson Tel: 800-959-2451 Fax: 805-296-5302 info@stgcase.com	A,U	MIS, TECH, RT	DT
BPWIN ERWIN/ERX	Logicworks	David Donovan 8245 Boone Blvd. Suite 400 Vienna, VA 22182 Tel: 703-761-1166 Fax: 703-761-1095		MIS, TECH	DT
CA-Estimacs	Computer Associates	Dana Williams 1 Computer Associates Plaza Islandia, NY 11788 Tel: 800-225-5224 Faax: 516-342-5734	O	MIS	DT
CA-Metrics	Computer Associates	Dana Williams 1 Computer Associates Plaza Islandia, NY 11788 Tel: 800-225-5224 Faax: 516-342-5734	A,U	MIS	DT
CA-Pan/LCM	Computer Associates	Dana Williams 1 Computer Associates Plaza Islandia, NY 11788 Tel: 800-225-5224 Faax: 516-342-5734	E	MIS, TECH, RT	MF
CA-Pan/Merge, CA-Panexec, CA- Panvalet	Computer Associates	Dana Williams 1 Computer Associates Plaza Islandia, NY 11788 Tel: 800-225-5224 Faax: 516-342-5734	E	All	MF
CA-Planmacs	Computer Associates	Dana Williams 1 Computer Associates Plaza Islandia, NY 11788 Tel: 800-225-5224 Faax: 516-342-5734	O	MIS	DT
CA-Realia, CA-Visual Realia	Computer Associates	Dana Williams 1 Computer Associates Plaza Islandia, NY 11788 Tel: 800-225-5224 Faax: 516-342-5734	E	TECH	DT
CA-Telon, CA-Telon PWS, CA-Visual Telon	Computer Associates	Dana Williams 1 Computer Associates Plaza Islandia, NY 11788 Tel: 800-225-5224 Faax: 516-342-5734	D,E	TECH	MF

*Appendix A: Upper CASE Products - Product List*

<b>Tool</b>	<b>Vendor</b>	<b>Contact Info</b>	<b>Type</b>	<b>Target</b>	<b>Platforms</b>
CA-Visual Objects	Computer Associates	Dana Williams 1 Computer Associates Plaza Islandia, NY 11788 Tel: 800-225-5224 Faax: 516-342-5734	D,C,E	MIS, TECH	DT
Canonizer	Sigma Six CASE	12456 S.E. 27th Place Suite 210 Bellevue, WA 98005 Tel: 800-827-4462 Fax: 206-641-7501 info@6signa.com	D	MIS	WS, DT
CDADL	Software Systems Design, Inc.	Tom Radi 909-625-6147	D, U		ALL
Chen Workbench	Chen & Associates, Inc.	Kirk Chedotal 4884 Constitution Ave Suite 1-E Baton Rouge, LA 70808 Tel: 504-928-5765 Fax: 504-928-9371	D, E	MIS, TECH	DT
DataViews	DataViews Corp.	Kevin Hassett Tel: 413-586-4144 Fax: 413-586-3805 kevin@dvcorp.com	A	RT	MF, WS
DocExpress	ATA, Inc	Doug Trolan 6701 Democracy Plaza Suite 710 Bethesda, MD 20817 Tel: 301-897-4101 Fax: 301-897-3106 dtrolan@cadre.com	U	All	WS, DT
DOORS	Zycad Corp	Susan Boers 11921 Freedom Drive Suite 550 Reston, VA 22090 Tel: 703-904-4360 Fax: 703-834-6622 susan_boers@zycad.com	A, U	All	DT, WS
EiffelCase	ISE	Darey Harrison 270 Storke Rd #7 Goleta, CA 93117 Tel: 805-685-1006 Fax: 805-685-6869 dareyh@eiffel.com	A, D, E	All	WS
Ensemble	Cadre Technologies	Doug Trolan 6701 Democracy Plaza Suite 710 Bethesda, MD 20817 Tel: 301-897-4101 Fax: 301-897-3106 dtrolan@cadre.com	D, E	TECH	WS
ERWIN/ERX	LogicWorks	David Donovan 8245 Boone Blvd. Suite 400 Vienna, VA 22182 Tel: 703-761-1166 Fax: 703-761-1095	A, D, E,	MIS, TECH	DT, WS

Tool	Vendor	Contact Info	Type	Target	Platforms
ES RE/Vision	Eden Systems Corp.	Brad Schweibold 9302 N. Meridian St. Suite 350 Indianapolis, IN 46260-1820 Tel: 317-848-9600 ext 224 Fax: 317-483-2271 custserv@iquest.net	A	MIS	DT, MF
Expert/CIO	P-Cube Corp.	Joseph Napoli 572 E. Lambert Road Brea, CA 92621 Tel 714-990-3169 Fax: 714-990-0838	A	MIS, RT	DT
firstcase	AGS Management Systems, Inc.	Valerie Palamountain 1012 W. Ninth Avenue King of Prussia, PA 19406 Tel: 800-220-2471 Fax: 610-265-1230	A,D,E	MIS,RT	DT
Foresight	Nu Thena Systems	Peter Stoupas 1430 Spring Hill Rd. Suite 200 McLean, VA 22102 Tel: 703-356-5056 Fax: 703-356-0498 info@nuthena.com	A,D	TECH, RT	WS
Foundation Design	Anderson Consulting Foundation	Mary Ryan 69 W. Washington St. Chicago, IL 60602 Tel: 800-458-8851	A,D,E,U	All	DT
Foundation Vista	Menlo Business Systems	Ronald Snow 201 Main Street Los Altos, CA 94022 Tel: 415-948-7920 Fax: 415-949-6655 menlomail@eWorld.com	A,D,U	All	DT
IE Advantage	Information Engineering Systems Corp.	Fred Krause 201 N. Union Street Alexandria, VA 22314 Tel: 703-739-2242 Fax: 703-739-0074	A,D,E,U	All	DT
Infomodeler	Asymetrix Corp.	Lance Wieland 110 110th Ave NE  Tel: 800-471-5184 x 2465 Fax: 206-637-1504 lancew@asymetrix.com		MIS	DT
iRAT	introspect Technologies, Inc.	Mary I Burns Cynthia M Mavros 1765 S. 8th Street Suite T8 Colorado Springs, CO 80906 Tel: 719-634-5744 Fax: 719-634-1163 intros@usa.net	A,D,E,U	TECH, RT	DT
LINC Environment	Unisys Corp.	Gary Olsen PO Box 500 Blue Bell, PA 19424 Tel: 215-986-3278 Fax: 215-986-6646 linc@p03.bb.unisys.com	D, C, E	MIS, TECH	WS

*Appendix A: Upper CASE Products - Product List*

<b>Tool</b>	<b>Vendor</b>	<b>Contact Info</b>	<b>Type</b>	<b>Target</b>	<b>Platforms</b>
MacAnalyst MacDesigner	Excel Software	Harold Halbleib PO Box 141 Marshalltown, IA 505018 Tel: 515-752-5359 Fax: 515-752-2435 casetools@aol.com	A, D, C, E, U	All	DT
Magec_RAD System	Magec Software	Vic Lee 1603 LBJ Freeway Suite 880 Dallas, TX 75234 Tel: 800-336-2432 Fax: 214-448-3030		MIS	ALL
McCabe Object- Oriented Tool	McCabe and Associates, Inc	Barbara Fiato 5501 Twin Knolls Rd #111 Columbia, MD 21045 Tel 800-638-6316 Fax: 410-995-1528 Mike@McCabe.com	A, D, C, E, U	TECH, RT	ALL
MethodMaker	Mark V Systems	Mo Bjornestad 16400 Ventura Blvd #300 Encino, CA 91436-2123 Tel: 818-995-7671 Fax: 818-995-4267 mob@markv.com	O	ALL	DT, WS
ObjectMaker, ObjectMaker TDK	Mark V Systems	Mo Bjornestad 16400 Ventura Blvd #300 Encino, CA 91436-2123 Tel: 818-995-7671 Fax: 818-995-4267 mob@markv.com	A, D, C, E, U, O	ALL	DT, WS
ObjectTeam	Cadre Technologies	Doug Trolan 6701 Democracy Plaza Suite 710 Bethesda, MD 20817 Tel: 301-897-4101 Fax: 301-897-3106 dtrolan@cadre.com	A, D, C, U, E	TECH, RT	ALL
ObjectTeam/ProDe v	Cadre Technologies	Doug Trolan 6701 Democracy Plaza Suite 710 Bethesda, MD 20817 Tel: 301-897-4101 Fax: 301-897-3106 dtrolan@cadre.com	D, C	TECH, MIS	WS
ObjectTeam for OMT	Cadre Technologies	Doug Trolan 6701 Democracy Plaza Suite 710 Bethesda, MD 20817 Tel: 301-897-4101 Fax: 301-897-3106 dtrolan@cadre.com	A, D, C U	ALL	WS, DT
Obsydian	Synon, Inc.	Byron Wilkes Tel: 206-223-1404 Fax: 206-382-9648	A, D, C, E, U	MIS	DT
PACE	Wang Laboratories	Judy Cole Tel: 508-967-2199 Fax: 508-967-9736 judy.cole@mailoff.wang.co m	O	MIS	WS, DT

*Software Technology Support Center*

<b>Tool</b>	<b>Vendor</b>	<b>Contact Info</b>	<b>Type</b>	<b>Target</b>	<b>Platforms</b>
Paradigm Plus	ProtoSoft	Kevin Shea Tel: 703-359-3755 Fax: 703-359-3753 shea@protosoft.com	A, D, C, E, U, O	ALL	DT, WS
ProcessMaker	Mark V Systems	Mo Bjornestad 16400 Ventura Blvd #300 Encino, CA 91436-2123 Tel: 818-995-7671 Fax: 818-995-4267 mob@markv.com	O	ALL	WS, DT
ProMod-AutoDesign	G & E Systems	Bryan Cooper 250 Edge Hill Rd. Sharon, MA 02067 Tel: 617-784-1007 Fax: 617-784-1737 bkocher@aol.com	D, C, U	ALL	ALL
ProMod-GUI, ProMod-Proface, ProMod-ProSQL	G & E Systems	Bryan Cooper 250 Edge Hill Rd. Sharon, MA 02067 Tel: 617-784-1007 Fax: 617-784-1737 bkocher@aol.com	D, C, O	ALL	ALL
ProMod-OCM	G & E Systems	Bryan Cooper 250 Edge Hill Rd. Sharon, MA 02067 Tel: 617-784-1007 Fax: 617-784-1737 bkocher@aol.com	O	ALL	WS
ProMod-Plus	G & E Systems	Bryan Cooper 250 Edge Hill Rd. Sharon, MA 02067 Tel: 617-784-1007 Fax: 617-784-1737 bkocher@aol.com	A, D, C, E, U, O	ALL	ALL
ProMod-SourcePilot	G & E Systems	Bryan Cooper 250 Edge Hill Rd. Sharon, MA 02067 Tel: 617-784-1007 Fax: 617-784-1737 bkocher@aol.com	D, C, O	ALL	WS
ProVision Workbench	Proforma Corp	Hugh Mensch 17515 West 9-Mile Rd. Southfield, MI 48075 Tel: 810-443-0506 Fax: 810-443-5495	A, U, E, O	MIS	DT
QASE-RT	Advanced System Technologies, Inc.	Don Hazell 12200 E. Briarwood Ave Suite 260 Englewood, CA 80401 Tel: 303-790-4242 x129 Fax: 303-790-2816 dhazell@advssystemtech.com	O	MIS, RT	DT, WS
Rational Rose	Rational Software Corp.	Loren Archer 2800 San Tomas Expressway Santa Clara, CA 95051 Tel: 408-496-3600 Fax: 408-496-3974 loren@rational.com	A,D,U,E	ALL	WS, DT

*Appendix A: Upper CASE Products - Product List*

<b>Tool</b>	<b>Vendor</b>	<b>Contact Info</b>	<b>Type</b>	<b>Target</b>	<b>Platforms</b>
Rational Rose/Ada, Rational Rose/C++	Rational Software Corp.	Loren Archer 2800 San Tomas Expressway Santa Clara, CA 95051 Tel: 408-496-3600 Fax: 408-496-3974 loren@rational.com	A, D, C, U, E	ALL	WS
RDBMS App Dev	Cadre Technologies	Doug Trolan 6701 Democracy Plaza Suite 710 Bethesda, MD 20817 Tel: 301-897-4101 Fax: 301-897-3106 dtrolan@cadre.com	A, O	MIS	WS, MF
ROBOCHART	Digital Insight	Lynn Morrison PO Box 533 Simi Valley, CA 93062 Tel: 805-583-3627 Fax: 805-583-3809 rc-sales@diginc.com	D, C, U, O	ALL	WS
RTM	Marconi System Technology	Distributor: Doug Trolan 6701 Democracy Plaza Suite 710 Bethesda, MD 20817 Tel: 301-897-4101 Fax: 301-897-3106 dtrolan@cadre.com	A, U	ALL	WS
SES/Objectbench	SES	Marc Fackler Tel: 719-522-9050 Fax: 719-522-9052 fackler@ses.com	D, C	ALL	WS
SES/Workbench	SES	Marc Fackler Tel: 719-522-9050 Fax: 719-522-9052 fackler@ses.com	A, D, C, O	ALL	WS
SoDA	Rational Software Corp.	Balaji Yelmannchili 2800 San Tomas Expressway Santa Clara, CA 95051 Tel: 408-496-3600 Fax: 408-496-3974 by@rational.com	D, C, U	ALL	WS
SoftTest	Bender & Associates, Inc.	Blaine Bragg PO Box 849 Larkspur, CA 94939 Tel: 415-924-9196 Fax: 415-924-3020 bbragg@softtest.com	A, T	ALL	DT, WS
SUMMIT-D, SUMMIT Process	Coopers & Lybrand L.L.P	John J. Newcomb Princeton Forrestal Village 136-300 Main Street Princeton, NJ 08540 Tel: 609-520-6131 Fax: 609-520-6195	E, U	MIS	DT
System Engineer	LBMS, Inc.	John Wills 1800 West Loop South 6th Floor Houston, TX 77029 Tel: 800-231-7575 Fax: 713-343-4419 Sales@lbms.com	A, D, C, E, U, O	MIS, TECH	DDT



Tool	Vendor	Contact Info	Type	Target	Platforms
Teamwork	Cadre Technologies, Inc.	Doug Trolan 6701 Democracy Plaza Suite 710 Bethesda, MD 20817 Tel: 301-897-4101 Fax: 301-897-3106 dtrolan@cadre.com	A, D, C, E, U, O	MIS, TECH	MF, MF
Teamwork/Ada	Cadre Technologies, Inc.	Doug Trolan 6701 Democracy Plaza Suite 710 Bethesda, MD 20817 Tel: 301-897-4101 Fax: 301-897-3106 dtrolan@cadre.com	D, C, O	TECH, RT	WS, MF
Teamwork/ Dynamic Analysis	Cadre Technologies, Inc.	Doug Trolan 6701 Democracy Plaza Suite 710 Bethesda, MD 20817 Tel: 301-897-4101 Fax: 301-897-3106 dtrolan@cadre.com	A	TECH, RT	WS, MF
Teamwork/ IMSQL	Cadre Technologies, Inc.	Doug Trolan 6701 Democracy Plaza Suite 710 Bethesda, MD 20817 Tel: 301-897-4101 Fax: 301-897-3106 dtrolan@cadre.com	A, O	MIS, TECH	WS, MF
Teamwork/RT	Cadre Technologies, Inc.	Doug Trolan 6701 Democracy Plaza Suite 710 Bethesda, MD 20817 Tel: 301-897-4101 Fax: 301-897-3106 dtrolan@cadre.com	A, D, U	TECH, RT	WS, MF
Teamwork/SA	Cadre Technologies, Inc.	Doug Trolan 6701 Democracy Plaza Suite 710 Bethesda, MD 20817 Tel: 301-897-4101 Fax: 301-897-3106 dtrolan@cadre.com	A, D, U	TECH, RT	WS, MF
Teamwork/SD	Cadre Technologies, Inc.	Doug Trolan 6701 Democracy Plaza Suite 710 Bethesda, MD 20817 Tel: 301-897-4101 Fax: 301-897-3106 dtrolan@cadre.com	D, C, O	TECH, RT	WS, MF
TurboCASE	StructSoft, Inc.	Shang Chyou 206-644-9834 206-644-7714 76636.254@compuserve.com	A, D, C, O, U	MIS, TECH, RT	DT
TurboCASE/Sys	StructSoft, Inc.	Shang Chyou 206-644-9834 206-644-7714 76636.254@compuserve.com	A,U	TECH, RT	DT

## **Appendix B: Upper CASE Product Sheets**

This appendix contains technology information sheets for most of the RAD tools in the tool lists. These reports provide more detailed information than the tools list. Users of these reports should be able to make preliminary tool assessments based on the provided information. Information on pricing, contact, life-cycle phases and activities, intended users, primary methodology base, hardware platforms, and general tool capabilities are included. The information has been divided into several sections. Following is a quick description of each section:

The **Product Information** and **Contact Information** sections should be self-explanatory.

The **Life Cycle Phases and Activities** section capture the intended purpose of the tool. Most tools are targeted at activities within specific phases of the software life cycle or are targeted at activities that occur across life cycle phases. The life cycle phases that are identified (Analysis, Design, Coding, and Maintenance) are generic and are not intended to imply any specific life cycle model such as waterfall or spiral. Re-engineering or reverse engineering tools are considered Maintenance tools. Several activities have been identified (e.g., Testing, Configuration Management, Documentation, and Project Management) that occur across life cycle phases. The list of activities is not meant to be all inclusive. Other activities are listed as "Other." Finally, one technology (Environments or Frameworks) that acts to connect tools that automate the various life cycle phases and activities is identified.

The **Intended Customers** section identifies the tool's intended market. Several examples of target customers such as MIS and Real-Time are listed. The "Other" category is for a tool that fills very specific market niche.

The **Primary Methodology** section addresses an issue important to the classification of analysis and design tools. Three complimentary and orthogonal views of software development methodologies have been recognized. They are object-oriented, process-oriented (structural or functional), and behavior-oriented (temporal, state-oriented, or dynamic). Each of these views takes a different perspective of the system being developed. Good development methodologies are typically concentric around one of the paradigms with support for the other two. Some tools may support multiple methodologies with differing primary paradigms.

The **Configurations** section lists the various Platforms/Operating-System on which the tool is currently available.

The **Description/Purpose** section contains free form text describing the tool.

The information in the reports was obtained either directly from the vendor or from the vendor's literature. In most cases, the vendor has supplied the information.

There are tools in the tool lists for which there is no associated technology information sheet. This condition occurs because there was insufficient available information to create the technology information sheet, either because the vendor did not supply information in time for publication or because the tool was added to the tool list too late for the creation of a technology information sheet.

The STSC can be contacted for both unpublished and updated reports that may be available.

## ADADL by Software Systems Design, Inc.

### Product Information:

Version Number: 5.6  
 Date of Last Release: 1995  
 Date of First Release: 1986  
 Number Sold: >300  
 Single User Price: From \$5,000

### Contact Information:

#### Point of Contact:

Name: Tom Radi  
 Address: 3627 Padua Ave.  
 Claremont, CA 91711  
 Phone : 909-625-6147

### Lifecycle Phases and Activities:

Requirements Specification	✓	High Level Design	✓	Re-engineering
Requirements Analysis	✓	Low Level Design		Environment
Requirements Documentation		Database		System Analysis
✓ Requirements Tracing/Mgmt.		Client-Server		Other:

### Intended Customers:

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	✓ Other: Ada Software Developers	

### Primary Methodology:

Structured	Object-Oriented	Behavior-Oriented
------------	-----------------	-------------------

### Configurations:

Unix (almost any workstation), DOS, VMS, PCs, Alpha

### Description/Purpose:

ADADL is an Ada/PDL used to design, document and analyze Ada programs.

ADADL, an Ada-based PDL fully satisfies the DoD Directive 3405.2 requirements for a compilable Ada/PDL. Using the ADADL processor improves both design quality and designer productivity.

ADADL analyzes both the pseudo-code and actual executable Ada code to detect logic errors and to produce a "pretty printed" output report which greatly simplifies understanding the design.

ADADL can be used for both forward and reverse engineering.

The analysis of the design results in over 25 additional reports that describe the design at any phase of development. Examples of design reports are cross reference of usage of Ada types, objects (including set/use information) and program units, invocation trees, declaration structure, with hierarchy, generic utilization, and interrupt reports.

The user can create up to 10 customized "project management" reports to identify such things as, requirements traceability (what requirements are satisfied by which program units), dates of completion of design reviews, coding or testing.

The design quality report shows areas of potentially questionable quality, such as where items are declared but never used or "with"ed but not needed.

The pseudo-code design and executable code are analyzed to calculate the McCabe complexity metric for each program unit, allowing overly complex program units to be identified early in the lifecycle.

The ADADL processor works in conjunction with the DocGen tool to automatically produce Mil/DoD Standard documentation, and with the TestGen tool to assist in design Unit test strategies.

## **Argos by Versant Object Technology**

### **Product Information:**

Version Number: Argos 1.4.2  
                   VERSANT ODBMS 3.0  
                   Smalltalk Interface 2.0.3  
 Date of Last Release: July 95  
 Date of First Release: June 93  
 Number Sold: --  
 Single User Price: \$3000

### **Contact Information:**

#### **Point of Contact:**

Name: Andy Smith  
 Address: 1380 Willow Rd  
           Menlo Park, CA 94025  
 Phone : 800-837-7268  
 Fax : 415-325-2380  
 E-mail : info@versant.com

### **Lifecycle Phases and Activities:**

Requirements Specification	✓ High Level Design	Re-engineering
✓ Requirements Analysis	✓ Low Level Design	Environment
Requirements Documentation	✓ Database	✓ System Analysis
Requirements Tracing/Mgmt.	✓ Client-Server	
✓ Other: Business Process Re-engineering		

### **Intended Customers:**

All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database		
✓ Other: Telecommunications, Transportation, Finance/Banking, Horizontal, Utilities, Healthcare, Network Management		

### **Primary Methodology:**

✓ Object-Oriented: Rumbaugh OMT, Jacobson Use-Case

### **Configurations:**

Solaris, OS/2, Sun OS, IBM/AIX, Windows, NT, Mac OS, HP/UX

### **Description/Purpose:**

The VERSANT ODBMS and VERSANT Argos™ provide an end-to-end solution for developing mission-critical, client-server applications. Developers model business requirements using graphical OOA&D tools while Argos executes the business model by synchronously generating Smalltalk applications. Argos also provides a tightly integrated visual programming facility for building portable GUI applications and reports. The VERSANT ODBMS serves as a design repository supporting team-based development and provides a robust, scalable object database management systems for multi-user distributed applications.

Application expertise: Telecommunication and network management; in particular, service creation and service delivery applications.

## ARIS/DesignGen by Software Systems Design, Inc.

### Product Information:

Version Number: 1.4  
 Date of Last Release: 1994  
 Date of First Release: 1991  
 Number Sold: 10-20  
 Single User Price: \$8,500

### Contact Information:

#### Point of Contact:

Name: Tom Radi  
 Address: 3627 Padua Ave.  
 Claremont, CA 91711  
 Phone : 909-625-6147

### Lifecycle Phases and Activities:

✓ Requirements Specification	✓ High Level Design	Re-engineering
✓ Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### Intended Customers:

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### Primary Methodology:

✓ Structured	Object-Oriented	Behavior-Oriented
--------------	-----------------	-------------------

### Configurations:

Works with TeamWork/RT, TeamWork/SA tools.

### Description/Purpose:

DesignGen (formerly called ARIS) is a set of software programs which automatically produce Ada code representing a Top Level Ada design. The design is constructed automatically by analyzing the software requirements as specified in a Structured Analysis database and creating a suitable "object oriented" Top Level Design using a combination of algorithms suggested by Booch, Goma and Shumate.

The DesignGen tool currently works in conjunction with the CADRE/Teamwork, IDE/Software Through Pictures and Excelerator.

The extracted software/system requirements are used to produce a preliminary Top Level Design. The requirements data base is analyzed to produce Ada packages which contain the Top Level Data structure and also to produce an Ada program which represents the Top Level Architecture. These Ada programs units are the starting point for the design process for further refinement by the design team.

The Ada software generated by DesignGen is compatible with any Ada compiler, and uses an Ada/PEL - ADADL (Ada Design and Documentation Language).

The TBD (To Be Determined) report is generated to help the design team to refine the design.

DesignGen provides a bridge between the RTSA requirements and the Ada design.

## ***Axiom-SA by STG, Inc.***

### **Product Information:**

Version Number: 4.2  
 Date of Last Release: June 1995  
 Date of First Release: November 1991  
 Number Sold: 1000  
 Single User Price: \$1995 single user  
 Quantity Discounts Available

### **Contact Information:**

#### **Point of Contact:**

Name: Vince Peterson  
 Address: 28157 Shelter Cove Dr.  
 Saugus, CA 91350  
 Phone : 800-959-2451/805-296-3607  
 Fax : 805-296-5302  
 E-mail : info@stgcase.com

### **Lifecycle Phases and Activities:**

✓ Requirements Specification	High Level Design	Re-engineering
✓ Requirements Analysis	Low Level Design	Environment
✓ Requirements Documentation	Database	System Analysis
✓ Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

✓ All	✓ Engineering	✓ Embedded Systems
✓ MIS	Client-Server	✓ Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured	Object-Oriented	Behavior-Oriented
--------------	-----------------	-------------------

### **Configurations:**

IBM PC and compatibles (MS-Windows)  
 Runs under emulators on Suns and Power Macs

### **Description/Purpose:**

Axiom-SA is a software requirements analysis tool supporting Structured Analysis with Hatley-Pirbhai real-time extensions. Requirements tracing is an integrated part of the tool. Automated documentation to meet all standards is supported. Axiom-SA is integrated with the AxiomDsn software design tool. STG is on the World Wide Web at <http://www.smartlink.net/~stgvjp>



## ***AxiomDsn by STG, Inc.***

### **Product Information:**

Version Number: 4.2  
Date of Last Release: June 1995  
Date of First Release: May 1995  
Number Sold: 50  
Single User Price: \$2995 single user  
Quantity Discounts Available

### **Contact Information:**

#### **Point of Contact:**

Name: Vince Peterson  
Address: 28157 Shelter Cove Dr.  
Saugus, CA 91350  
Phone : 800-959-2451/805-296-3607  
Fax : 805-296-5302  
E-mail : info@stgcase.com

### **Lifecycle Phases and Activities:**

Requirements Specification	√	High Level Design	Re-engineering
Requirements Analysis	√	Low Level Design	Environment
Requirements Documentation		Database	System Analysis
√ Requirements Tracing/Mgmt.		Client-Server	Other:

### **Intended Customers:**

All	√	Engineering	√	Embedded Systems
√ MIS		Client-Server	√	Real-time
Database		Other:		

### **Primary Methodology:**

√ Structured      √ Object-Oriented      Behavior-Oriented

### **Configurations:**

IBM PC and compatibles (MS-Windows)  
Runs under emulators on Suns and Power Macs

### **Description/Purpose:**

AxiomDsn is a software design tool to support the design of software design in procedural languages such as Ada and C. It provides syntax checking, comprehensive automated documentation to meet all design specifications requirements, include traceability, McCabe cyclometric complexity calculations and code frame generation. AxiomDsn is integrated with the Axiom-SA and AxiomSys analysis tools. STG is on the World Wide Web at <http://www.smartlink.net/~stgvjp>

## **AxiomSys by STG, Inc.**

### **Product Information:**

Version Number: 4.2  
 Date of Last Release: June 1995  
 Date of First Release: May 1994  
 Number Sold: 500  
 Single User Price: \$2495 single user  
 Quantity Discounts Available

### **Contact Information:**

#### **Point of Contact:**

Name: Vince Peterson  
 Address: 28157 Shelter Cove Dr.  
 Saugus, CA 91350  
 Phone : 800-959-2451/805-296-3607  
 Fax : 805-296-5302  
 E-mail : info@stgcase.com

### **Lifecycle Phases and Activities:**

✓ Requirements Specification	High Level Design	Re-engineering
✓ Requirements Analysis	Low Level Design	Environment
✓ Requirements Documentation	Database	✓ System Analysis
✓ Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

All	✓ Engineering	✓ Embedded Systems
✓ MIS	Client-Server	✓ Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured	Object-Oriented	Behavior-Oriented
--------------	-----------------	-------------------

### **Configurations:**

IBM PC and compatibles (MS-Windows)  
 Runs under emulators on Suns and Power Macs

### **Description/Purpose:**

AxiomSys is a system requirements analysis tool supporting Structured Analysis with Hatley-Pirbhai real-time extensions and the Hatley-Pirbhai Architecture Modeling. Requirements tracing is an integrated part of the tool. Automated documentation to meet all standards is supported. AxiomSys is integrated with the AxiomDsn Software design tool. STG is on the World Wide Web at <http://www.smartlink.net/~stgvjp>

## ***BPWIN by LogicWorks***

### **Product Information:**

Version Number: 1.5.1  
Date of Last Release: May 1995  
Date of First Release: August 1993  
Number Sold: 5,000  
Single User Price: \$2,495 Single User  
\$1,996 GSA Single User

### **Contact Information:**

#### **Point of Contact:**

Name: David Donovan  
Address: 8245 Boone Blvd  
Suite 400  
Vienna, VA 22182  
Phone : 703-761-1166  
Fax : 703-761-1095  
E-mail :

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

All	✓ Engineering	Embedded Systems
MIS	✓ Client-Server	Real-time
✓ Database	Other:	

### **Primary Methodology:**

✓ Structured      ✓ Object-Oriented      Behavior-Oriented

### **Configurations:**

MS-Windows

### **Description/Purpose:**

BPWIN is an IDEF0 Business Process Reengineering tool which provides the ability to produce graphical representations of a business process which will enable your organization to find areas of improvement, raise level of quality and lower cost. ERWIN/ERX and BPWIN cross validate business data and business process models.

## CA-Estimacs by Computer Associates International, Inc.

### Product Information:

Version Number: 7.1  
 Date of Last Release: February, 1995  
 Date of First Release: 1982  
 Number Sold:  
 Single User Price: Contact Vendor

### Contact Information:

#### Point of Contact:

Name: Dana Williams  
 Address: One Computer Associates Plaza  
 Islandia, NY 11788  
 Phone : 800-225-5224  
 Fax : 516-342-5734  
 E-mail :

### Lifecycle Phases and Activities:

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	
√ Other: Project Estimation		

### Intended Customers:

√ All	Engineering	Embedded Systems
√ MIS	Client-Server	Real-time
Database	Other: Application developers	

### Primary Methodology:

√ Structured      √ Object-Oriented      Behavior-Oriented

### Configurations:

IBM PC or compatible (MS-DOS)

### Description/Purpose:

CA-Estimacs is an estimating model using research drawn from a database of more than 18,000 completed software projects. CA-Estimacs is designed to provide software developers with time/cost/resources estimates (within +/- 15% of actual) during the feasibility phase of the life cycle. CA-Estimacs delivers estimates without requiring the user to understand function points or SLOC. In addition, CA-Estimacs also provides a Risk Analysis model for assessing a project's vulnerability; A Financial Model for determining cost and benefits of a project; An estimate of the number of function points a project will deliver (based on the effort estimate); and a Portfolio Analysis Component that enables you to conduct strategic planning to determine whether the system meets your business goals and also determine the resource requirements needed across multiple projects.

STSC RED Product Sheet Version 2.0c

## **CA-Metrics by Computer Associates International, Inc.**

### **Product Information:**

Version Number: 3.0  
 Date of Last Release: January, 1995  
 Date of First Release: 1992  
 Number Sold:  
 Single User Price: Contact Vendor

### **Contact Information:**

#### **Point of Contact:**

Name: Dana Williams  
 Address: 1 Computer Associates Plaza  
 Islandia, NY 11788  
 Phone : 800-225-5224  
 Fax : 516-342-5734  
 E-mail :

### **Lifecycle Phases and Activities:**

√ Requirements Specification	High Level Design	Re-engineering
√ Requirements Analysis	Low Level Design	Environment
√ Requirements Documentation	Database	System Analysis
√ Requirements Tracing/Mgmt.	Client-Server	
√ Other: Measures development efforts		

### **Intended Customers:**

√ All	Engineering	Embedded Systems
√ MIS	Client-Server	Real-time
Database	Other: Application developers	

### **Primary Methodology:**

√ Structured	Object-Oriented	Behavior-Oriented
--------------	-----------------	-------------------

### **Configurations:**

IBM PC or compatible (MS-DOS)

### **Description/Purpose:**

CA-Metrics is a measurement and tracking system for IS organizations. It allows users to define key performance measures, like actual cost versus estimated cost, time to delivery, defect ratios, and maintenance costs ratios, for monitoring and improving performance. CA-Metrics includes the Total Quality Management/Software tool (TQM/S) for tracking and analyzing problems and defects. TQM/S provides detailed reports useful for root cause analysis of defects, and pinpointing the phase of the life cycle which caused the problem. CA-Metrics includes function point analysis, lines of code analysis, benchmark date, and import and export capabilities to or from other project tracking, time reporting, estimation or defect tracking tools such as CA-Estimacs, CA-FPXpert, and CA-SuperProject. Graphical and tabular displays and reports are also provided.

STSC RED Product Sheet Version 2.0c

**CA-Pan/LCM Configuration Manager**  
*by*  
**Computer Associates International, Inc.**

**Product Information:**

Version Number: 1.0  
Date of Last Release: January 1994  
Date of First Release: 1993  
Number Sold:  
Single User Price: Contact Vendor

**Contact Information:**

**Point of Contact:**

Name: Dana Williams  
Address: One Computer Associates Plaza  
Islandia, NY 11788  
Phone : 800-225-5224  
Fax : 516-342-5734  
E-mail :

**Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	✓ Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

**Intended Customers:**

✓ All	✓ Engineering	✓ Embedded Systems
✓ MIS	✓ Client-Server	✓ Real-time
Database	Other:	

**Primary Methodology:**

✓ Structured                      Object-Oriented                      Behavior-Oriented

**Configurations:**

IBM Mainframe (MVS, VSE)

**Description/Purpose:**

CA-Pan/LCM Configuration Manager is a comprehensive configuration management system for the mainframe. CA-Pan/LCM Configuration Manager determines the dependencies between application components and automates the process of recompiling, reassembling, and relinking the application software system. CA-Pan/LCM Configuration Management also provides the impact analysis information necessary for project planning and management. This product can be used in conjunction with other products in the CA Life Cycle Management Suite.

## **CA-Pan/Merge by Computer Associates International, Inc.**

### **Product Information:**

Version Number: 1.0  
Date of Last Release: 1990  
Date of First Release: 1990  
Number Sold:  
Single User Price: Contact Vendor

### **Contact Information:**

#### **Point of Contact:**

Name: Dana Williams  
Address: 1 Computer Associates Plaza  
Islandia, NY 11788  
Phone : 800-225-5224  
Fax : 516-342-5734  
E-mail :

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	✓ Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured	Object-Oriented	Behavior-Oriented
--------------	-----------------	-------------------

### **Configurations:**

IBM Mainframe (MVS)

### **Description/Purpose:**

CA-Pan/Merge is a development tool that allows users to easily combine separate sets of program changes into one program. CA-Pan/Merge automatically identifies all changes and overlays. Any conflicting changes are clearly marked and can easily be resolved. Conflicting changes can be resolved online. CA-Pan/Merge provides both detailed and summary reports. Its ISPF panel-driven approach makes it easy to use. CA-Pan/Merge simplifies the control of concurrent software development projects. This product can be used in conjunction with other products in the CA Life Cycle Management Suite.

## CA- Panexec by Computer Associates International, Inc.

### Product Information:

Version Number: 5.3  
 Date of Last Release: May, 1994  
 Date of First Release: 1977  
 Number Sold:  
 Single User Price: Contact Vendor

### Contact Information:

#### Point of Contact:

Name: Dana Williams  
 Address: 1 Computer Associates Plaza  
 Islandia, NY 11788  
 Phone : 800-225-5224  
 Fax : 516-342-5734  
 E-mail :

### Lifecycle Phases and Activities:

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	✓ Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### Intended Customers:

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### Primary Methodology:

✓ Structured	Object-Oriented	Behavior-Oriented
--------------	-----------------	-------------------

### Configurations:

IBM Mainframe (MVS)

### Description/Purpose:

CA-Panexec is a software system that centralizes the storage, management and control for executable and object members. Other data types may also be stored in a CA-Panexec library. CA-Panexec provides flexible security and control over important programs in both test and production status. CA-Panexec eliminates library reorganization and provides automatic DASD error recovery. An ISPF interface is provided for ease of use. A cross reference of source to executable is available which supports CA-Panvalet, CA-Librarian, and PDS source libraries. This product can be used in conjunction with other products in the CA Life Cycle Management Suite.



## **CA-Panvalet by Computer Associates International, Inc.**

### **Product Information:**

Version Number: 14.2  
 Date of Last Release: February 1994  
 Date of First Release: 1969  
 Number Sold:  
 Single User Price: Contact Vendor

### **Contact Information:**

#### **Point of Contact:**

Name: Dana Williams  
 Address: 1 Computer Associates Plaza  
 Islandia, NY 11788  
 Phone : 800-225-5224  
 Fax : 516-342-5734  
 E-mail :

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	√ Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other: *

### **Intended Customers:**

√ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

√ Structured	Object-Oriented	Behavior-Oriented
--------------	-----------------	-------------------

### **Configurations:**

IBM Mainframe (MVS, VSE)

### **Description/Purpose:**

CA-Panvalet is a library management system that centralizes the storage of source, JCL, object and data. Programmers benefit from member locking, a powerful compare program, and a comprehensive directory. Managers can take advantage of access controls including status codes to provide protection of production members, a reporting facility, and automatic archival procedures. Efficient DASD utilization is also ensured through dynamic space allocation and member compression. Online functionality is provided with TSO, ISPF, CMS and ICCF options. This product can be used in conjunction with other products in the CA Life Cycle Management Suite.

## ***CA-Planmacs by Computer Associates International, Inc.***

### **Product Information:**

Version Number: 2.1  
 Date of Last Release: January, 1995  
 Date of First Release: 1984  
 Number Sold:  
 Single User Price: Contact Vendor

### **Contact Information:**

#### **Point of Contact:**

Name: Dana Williams  
 Address: One Computer Associates Plaza  
 Islandia, NY 11788  
 Phone : 800-225-5224  
 Fax : 516-342-5734  
 E-mail :

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	
√ Other: Project Planning		

### **Intended Customers:**

√ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other: Application developers	

### **Primary Methodology:**

√ Structured                      Object-Oriented                      Behavior-Oriented

### **Configurations:**

IBM PC or compatible (MS-DOS)

### **Description/Purpose:**

CA-Planmacs is a knowledge-based development modeling/planning system that quickly generates full-scale application development plans that would normally take days or weeks of manual effort. CA-Planmacs interface directly with CA-Estimacs to provide project managers with a detailed project plan based on a work breakdown structure. With this information, manager scan quickly determine their needs down to the task and activity levels. CA-Planmacs also creates export files for popular project management systems, eliminating a great deal of time-consuming data entry.

**CA-Realia II Workbench by Computer Associates International, Inc.**

**Product Information:**

Version Number: 1.1, 1.2  
Date of Last Release: May, 1995  
Date of First Release: 1993  
Number Sold:  
Single User Price: Contact Vendor

**Contact Information:**

**Point of Contact:**

Name: Dana Williams  
Address: 1 Computer Associates Plaza  
Islandia, NY 11788  
Phone : 800-225-5224  
Fax : 516-342-5734  
E-mail :

**Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	✓ Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	✓ Client-Server	
✓ Other: Character-based COBOL application development.		

**Intended Customers:**

All	Engineering	Embedded Systems
MIS	✓ Client-Server	Real-time
Database	✓ Other: Application developers	

**Primary Methodology:**

✓ Structured      Object-Oriented      Behavior-Oriented

**Configurations:**

IBM PC & Compatible (Windows, OS/2)

**Description/Purpose:**

CA-Realia II Workbench is a complete COBOL development system that is hosted by Windows or OS/2 Presentation Manager. It is an integrated, multi-tasking graphical environment that includes a high-speed COBOL compiler, a Windows-hosted, source-level debugger, a COBOL-intelligent analyzer, a CICS emulator, and integrated GUI editor and a complete life cycle management tool for source control, a VSAM emulator, and support for most PC file types and databases. Options use the Workbench as a front end for analyzing, testing, and debugging mainframe applications remotely.

**CA-Telon by Computer Associates International, Inc.**

**Product Information:**

Version Number: 2.3  
 Date of Last Release: October, 1994  
 Date of First Release: 1981  
 Number Sold:  
 Single User Price: Contact Vendor

**Contact Information:**

**Point of Contact:**

Name: Dana Williams  
 Address: 1 Computer Associates Plaza  
 Islandia, NY 11788  
 Phone : 800-225-5224  
 Fax : 516-342-5734  
 E-mail :

**Lifecycle Phases and Activities:**

Requirements Specification	√ High Level Design	Re-engineering
Requirements Analysis	√ Low Level Design	√ Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

**Intended Customers:**

All	√ Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other: Application developers	

**Primary Methodology:**

√ Structured	Object-Oriented	Behavior-Oriented
--------------	-----------------	-------------------

**Configurations:**

IBM 370, 30xx, 43xx or compatible (MVS)

**Description/Purpose:**

CA-Telon is a complete application development system that reduces the time required to design, code, test, debug, document, and maintain applications. It supports design prototyping, rules-based generation, and testing for program development and maintenance on the mainframe or workstation. It generates COBOL, COBOL II or PL/1 code for batch, CICS, IMS/DC, and VSE, and automatically generates data access for CA-Datcom/SQL Option, CA-IDMS/SQL option, DB2 and IMS databases plus VSAM and sequential file processing. CA-Telon also generates COBOL for AS/400, UNIX, PC-DOS, and OS/2 with SQL data access generated for a variety of relational databases.

**CA-Telon PWS by Computer Associates International, Inc.**

**Product Information:**

Version Number: 2.3  
 Date of Last Release: October, 1994  
 Date of First Release: 1989  
 Number Sold:  
 Single User Price: Contact Vendor

**Contact Information:**

**Point of Contact:**

Name: Dana Williams  
 Address: 1 Computer Associates Plaza  
 Islandia, NY 11788  
 Phone : 800-225-5224  
 Fax : 516-342-5734  
 E-mail :

**Lifecycle Phases and Activities:**

Requirements Specification	✓	High Level Design		Re-engineering
Requirements Analysis	✓	Low Level Design	✓	Environment
Requirements Documentation		Database		System Analysis
Requirements Tracing/Mgmt.		Client-Server		Other:

**Intended Customers:**

All	✓	Engineering	Embedded Systems
MIS		Client-Server	Real-time
Database		Other: Application developers	

**Primary Methodology:**

✓ Structured                      Object-Oriented                      Behavior-Oriented

**Configurations:**

IBM 386 and above or compatible with PC-DOS or OS/2 and support for multi-user LAN implementations under Novell NetWare or OS/2 LAN Server.

**Description/Purpose:**

CA-Telon PWS is a complete implementation of the mainframe version of CA-Telon designed to run on MS-DOS and OS/2. With CA-Telon for the Programmable Work Station (PWS), users can offload their application development projects from the mainframe and provide programmers with a complete application development environment on the workstation. CA-Telon PWS generates COBOL, COBOL II, or PL/1 source programs that access CA-IDMS/DB, IMS, VSAM, and a variety of RDBMS's. The CA-Telon Design Facility may be used for designing and prototyping on-line and batch applications. The CA-Telon Generator facility is used for generating structured portable code for AS/400, IMS, CICS, UNIX, Tandem, MVS, VSE, OS/2 and PC-DOS. CA-Telon PWS works with either CA-Realia or Micro Focus COBOL compilers, debuggers or test facilities. With CA-Telon PWS, developers can have the full advantages of working on the workstation without losing compatibility with the mainframe. All new development and modification of existing CA-Telon created programs can be done on the workstation with CA-Telon PWS then stored and combined with existing systems on the mainframe.

**CA-Visual Objects by Computer Associates International, Inc.**

**Product Information:**

Version Number: 1.0  
 Date of Last Release: June 1995  
 Date of First Release: December 1994  
 Number Sold:  
 Single User Price: Contact Vendor

**Contact Information:**

**Point of Contact:**

Name: Dana Williams  
 Address: 1 Computer Associates Plaza  
 Islandia, NY 11788  
 Phone : 800-225-5224  
 Fax : 516-342-5734  
 E-mail :

**Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	√ Re-engineering
Requirements Analysis	√ Low Level Design	√ Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	√ Client-Server	Other:

**Intended Customers:**

√ All	√ Engineering	Embedded Systems
√ MIS	√ Client-Server	Real-time
√ Database	Other:	

**Primary Methodology:**

√ Structured      √ Object-Oriented      √ Behavior-Oriented

**Configurations:**

486 Class machine (Windows)

**Description/Purpose:**

CA-Visual Objects™ is a robust, extensible, object-oriented, native code compiler integrated with a GUI development environment and replaceable database drivers that breaks new ground in PC applications development. Based on an active object-oriented repository, it provides both the ease-of-use and fast prototyping capabilities of visual programming environments, and the performance, flexibility and power found only in a true compiler. It provides the ideal platform for development of enterprise-wide mission-critical applications.

## **CA-Visual Realia by Computer Associates International, Inc.**

### **Product Information:**

Version Number: 1.0  
 Date of Last Release: 1995  
 Date of First Release: 1994  
 Number Sold:  
 Single User Price: Contact Vendor

### **Contact Information:**

#### **Point of Contact:**

Name: Dana Williams  
 Address: One Computer Associates Plaza  
 Islandia, NY 11788  
 Phone : 800-225-5224  
 Fax : 516-342-5734  
 E-mail :

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	√ Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	√ Client-Server	
√ Other: COBOL Application Development for windows		

### **Intended Customers:**

All	Engineering	Embedded Systems
MIS	√ Client-Server	Real-time
Database	√ Other: Application Developers	

### **Primary Methodology:**

√ Structured      Object-Oriented      Behavior-Oriented

### **Configurations:**

IBM PC and compatible (Windows)

### **Description/Purpose:**

CA-Visual Realia is a complete Windows development that provides COBOL programmers with the tools needed to create GUI client/server applications. CA-Visual Realia uses COBOL for the business logic and GUI Windows tools for the user interface. CA-Visual Realia facilitates the development of new GUI corporate applications while allowing legacy systems to acquire a GUI front-end.

## CA-Visual Telon by Computer Associates International, Inc.

### Product Information:

Version Number: 2.3  
 Date of Last Release: October, 1994  
 Date of First Release: 1989  
 Number Sold:  
 Single User Price: Contact Vendor

### Contact Information:

#### Point of Contact:

Name: Dana Williams  
 Address: 1 Computer Associates Plaza  
 Islandia, NY 11788  
 Phone : 800-225-5224  
 Fax : 516-342-5734  
 E-mail :

### Lifecycle Phases and Activities:

Requirements Specification	✓ High Level Design	Re-engineering
Requirements Analysis	✓ Low Level Design	✓ Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### Intended Customers:

All	✓ Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other: Application developers	

### Primary Methodology:

✓ Structured                      Object-Oriented                      Behavior-Oriented

### Configurations:

IBM 386 and above or compatible with PC-DOS or OS/2 and support for multi-user LAN implementations under Novell NetWare or OS/2 LAN Server.

### Description/Purpose:

CA-Visual Telon is a fully integrated GUI client/server application generation tool for the development of enterprise level distributed applications. It delivers sophisticated GUI applications for end users and provides developers with a choice of architectures for distributed applications in client/server environments. The components include a project manager, program painter, prototyping facility, COBOL code generator, data administration, local database, compiler, COBOL analyzer, source code debugger, reporter, and a life cycle manager. CA-Visual Telon is designed to let CA-Telon clients leverage their investments in existing applications, development and training through use of a migration facility. This migration facility translates existing CA-Telon screen definition programs into new CA-Visual Telon graphical models that are used to generate GUI client/.server programs.

STSC RED Product Sheet Version 2.0c



**Canonizer by Sigma Six CASE, Inc.**

**Product Information:**

Version Number: 2.0  
Date of Last Release: 1995  
Date of First Release: 1989  
Number Sold:  
Single User Price: \$2,995 - \$4,995  
Multiple user license discount available.

**Contact Information:**

**Point of Contact:**

Name:  
Address: 13456 S.E. 27th Place  
Suite 210  
Bellevue, WA 98005  
Phone : 800-827-4462  
Fax : 206-641-7501  
E-mail : info@6sigma.com

**Lifecycle Phases and Activities:**

Requirements Specification	✓ High Level Design	Re-engineering
Requirements Analysis	Low Level Design	Environment
Requirements Documentation	✓ Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	
Other:		

**Intended Customers:**

All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
✓ Database	Other: Developers	

**Primary Methodology:**

Structured	Object-Oriented	Behavior-Oriented
------------	-----------------	-------------------

**Configurations:**

IMB-PC (DOS, Windows), Sun Sparc (Sun OS 4.1, Solaris 1.0,2.1), IBM RS/6000(AIX), DEC RISC(ULTRIX), HP 9000 (HP-UX).

**Description/Purpose:**

The Canonizer is best suited to the development of custom data bases having a high level of complexity, particularly medium to large data bases with many inter-relationships and high numbers of tables and data items. With the Canonizer, a developer can design or redesign a data base by creating data models using Attribute Relationship Diagrams (ARDs). ARDs map the relationships between attributes with a table and step-by-step process that provides robust, detailed data models.

Automatic normalization, to the Third Normal Form, allows rapid prototyping and testing, and the Canonizer's normalized models can be accessed and denormalized as necessary. The Canonizer also maintains a data item dictionary, and produces a variety of useful reports and documentation.

For data base migration, the Canonizer imports data models and export DBMS-specific code, Data base supported include ANSI SQL, DB2, Informix, Ingres, Oracle, /rdb, SQL Server, Sybase, Unify and XDB. CASE tools supported are Cadre Teamwork and IDE Software through Pictures; programming structures include Ada, C, COBOL and Pascal.

## **CDADL by Software Systems Design, Inc.**

### **Product Information:**

Version Number: 3.2  
 Date of Last Release: 1995  
 Date of First Release: 1992  
 Number Sold: 20+  
 Single User Price: From \$3,000

### **Contact Information:**

#### **Point of Contact:**

Name: Tom Radi  
 Address: 3627 Padua Ave.  
 Claremont, CA 91711  
 Phone : 909-625-6147  
 Fax :  
 E-mail :

### **Lifecycle Phases and Activities:**

Requirements Specification	✓	High Level Design	✓	Re-engineering
Requirements Analysis	✓	Low Level Design		Environment
Requirements Documentation		Database		System Analysis
✓ Requirements Tracing/Mgmt.		Client-Server		Other:

### **Intended Customers:**

All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

Structured	Object-Oriented	Behavior-Oriented
------------	-----------------	-------------------

### **Configurations:**

Unix (almost any workstation), DOS, VMS

### **Description/Purpose:**

CDADL, a C-based PDL improves both design quality and designer productivity.

CDADL analyzes both the pseudo-code and actual executable C code to detect logic errors and to produce output reports which greatly simplifies understanding the design.

The analysis of the design results in over 25 additional reports that describe the design at any phase of development. Examples of design reports are cross reference of usage of types, objects (including set/use information) and functions, invocation trees, declaration structure (file contents).

The user can create up to 10 customized "project management" reports to identify such things as, requirements traceability, dates of completion of design reviews, coding or testing. The design quality reports shows areas of potentially questionable quality, such as where items are declared but never used.

The pseudo-code design and executable code are analyzed to calculate the McCabe complexity metric for each function, allowing overly complex programs to identified early in the lifecycle.

The CDADL processor works in conjunction with the DocGen-C tool to automatically produce Mil/DOD standard documentation, and with the TestGen-C tool to assist in developing test strategies for each function, or to review the pseudo-code.

The CDADL processor is available on a wide variety of hosts, including IBM RS6000, VAX/VMS Unix or Ultrix, Apollo, Sun, HP9000, Data General and 80386/DOS and Unix.

## ***Chen Workbench by Chen & Associates, Inc.***

### **Product Information:**

Version Number:  
Date of Last Release:  
Date of First Release:  
Number Sold:  
Single User Price:

### **Contact Information:**

#### **Point of Contact:**

Name: Kirk Chedotal  
Address: 4884 Constitution Ave  
Suite 1-E  
Baton Rouge, LA 70808  
Phone : 504-928-5765  
Fax : 504-928-9371

### **Lifecycle Phases and Activities:**

Requirements Specification	✓	High Level Design	✓	Re-engineering
Requirements Analysis		Low Level Design		Environment
Requirements Documentation	✓	Database		System Analysis
Requirements Tracing/Mgmt.	✓	Client-Server		Other:

### **Intended Customers:**

✓ All	✓ Engineering	Embedded Systems
✓ MIS	✓ Client-Server	Real-time
✓ Database	Other:	

### **Primary Methodology:**

✓ Structured      Object-Oriented      Behavior-Oriented

### **Configurations:**

PC or compatible (DOS, Windows)

### **Description/Purpose:**

The Chen ER-Modeler Workbench provides CASE tools supporting data and process modeling, normalization, dictionary, schema generation, and reverse engineering. Migration/conversion is also supported. Chen diagrams use two different Entity-Relationship notation; Process models use either Gane-Sarson or Yourdon-DeMarco notation. The Chen Normalizer is one of the most sophisticated on the market, performing Normalization to the third normal form. Functional dependencies can be saved to facilitate denormalization. Links to major data dictionaries and CASE tools are provided. ER-Modeler is available in either network or standalone versions and run on a PC or compatible with DOS or Windows and a graphics cards.

## **DataViews by DataVeivs Corp.**

### **Product Information:**

Version Number: 9.7  
 Date of Last Release: July 1995  
 Date of First Release: 1986  
 Number Sold: 18,000  
 Single User Price: \$17,700

### **Contact Information:**

#### **Point of Contact:**

Name: Kevin Hassett  
 Address: 47 Pleasant Street  
 Northampton, MA 01060  
 Phone : 413-586-4144  
 Fax : 413-586-3805  
 E-mail : kevin@dvcorp.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
✓ Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	
✓ Other: Process Technologies		

### **Intended Customers:**

All	Engineering	Embedded Systems
MIS	Client-Server	✓ Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured                      Object-Oriented                      ✓ Behavior-Oriented

### **Configurations:**

Sun OS, Solaris, Alpha Open VMS, DEC UNIX, SCO, SGI, IBM RS/6000, HP 9000

### **Description/Purpose:**

DataViews lets you deliver high-end, animated graphics to your customer. DataViews lets you build operator interfaces, soft instruments, and market-specific editors to enhance your product. Whether you're building DCS systems, PLC interfaces, or front ends to SCADA. DataViews gives you a maintained, off-the-shelf graphics development tool. DataViews eliminates the cost and risk of developing and maintaining internal proprietary graphics libraries. Furthermore, DataViews graphics are portable across UNIX, VMS, Windows-NT and Windows. DataViews gives you an object-based drawing editor and a subroutine library. The point and click drawing editor lets you draw a visual representation of your world, whether it contains tanks, pipes, and valves, or planes, vehicles, and airports. The DataViews subroutine library is a collection of high-level graphics routines that let you manipulate graphical objects, redirect their data sources, and attach functions to the objects.

## ***DocEXPRESS by ATA, Inc.***

### **Product Information:**

Version Number: 2.0  
Date of Last Release:  
Date of First Release:  
Number Sold:  
Single User Price: Discounts Available

### **Contact Information:**

#### **Point of Contact:**

Name: Doug Trolan, Dir Fed Ops  
Address: 6701 Democracy Plaza  
Suite 710  
Bethesda, MD 20817  
Phone : 301-897-4101  
Fax : 301-897-3106  
E-mail : dtrolan@cadre.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	Environment
✓ Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured      Object-Oriented      ✓ Behavior-Oriented

### **Configurations:**

Sun (SUN OS 4.1.3), Sun (Solaris 2.\*), HP (HPUX), IBM RS6000 (AIX) and PC (Windows 3.1).

### **Description/Purpose:**

A robust document production capability with extensive fill-in guides (templates) for DOD standard documents. Extensive advice on how to map software engineering artifacts to specific sections of DOD document standards including MIL-STD-2167A and MIL-STD-498 are included. DocEXPRESS operates with Interleaf or FrameMaker. DocEXPRESS comes with an interface to Cadre's Teamwork and ObjectTeam families of CASE tools for software system engineering.

## ***DOORS by Zycad Corp.***

### **Product Information:**

Version Number: 2.1.5  
 Date of Last Release: January 1995  
 Date of First Release: 1990  
 Number Sold: >1000  
 Single User Price: Contact Vendor

### **Contact Information:**

#### **Point of Contact:**

Name: Susan Boers  
 Address: 11921 Freedom Drive  
 Suite 200  
 Reston, VA 22090  
 Phone : 703-904-4360  
 Fax : 703-834-6622  
 E-mail : susan\_boers@zycad.com

### **Lifecycle Phases and Activities:**

√ Requirements Specification	High Level Design	Re-engineering
√ Requirements Analysis	Low Level Design	Environment
√ Requirements Documentation	Database	System Analysis
√ Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

√ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

Structured	Object-Oriented	Behavior-Oriented
------------	-----------------	-------------------

### **Configurations:**

Sun (SUN OS 4.1.3), Sun (Solaris 2.\*), HP (HPUX), IBM RS6000 (AIX) and PC (Windows 3.1).

### **Description/Purpose:**

Full lifecycle requirements engineering/traceability/management toolset. Imports almost any file formats, produces verification matrices, impact analyses, exception reports, etc. and outputs reports in almost any format, includes interfaces to other tools.

## *EiffelCase by ISE*

### Product Information:

Version Number: 3.3  
 Date of Last Release: August 1995  
 Date of First Release: 1991  
 Number Sold:  
 Single User Price: \$1,995 -

### Contact Information:

#### Point of Contact:

Name: Darey Harrison  
 Address: 270 Storke Rd #7  
 Goleta, CA 93117  
 Phone : 805-685-1006  
 Fax : 805-685-6869  
 E-mail : dareyh@eiffel.com

### Lifecycle Phases and Activities:

Requirements Specification	✓	High Level Design	✓	Re-engineering
Requirements Analysis	✓	Low Level Design	✓	Environment
Requirements Documentation		Database	✓	System Analysis
Requirements Tracing/Mgmt.	✓	Client-Server		
Other:				

### Intended Customers:

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other: Developers	

### Primary Methodology:

Structured      ✓ Object-Oriented      Behavior-Oriented

### Configurations:

SunOS, Solaris, Linux, HP9000, Silicon Graphics, DEX/Alpha, Windows (NT, 3.1, '95), SCO-ODT, IBM RS6000.

### Description/Purpose:

EiffelCase is an object-oriented analysis & design workbench and CASE tool. It covers the full life cycle of software modeling, architectural design and user-developer communication. It assists in preparing, documenting and tracking identified object-oriented system elements and their relationships from preliminary specifications to the programming phase. EiffelCase helps to build and display multiple views: type charts, graphics, formal text. It guarantees inter-view compatibility through automatic update procedures. Based on BON (Better Object Notation), EiffelCase provides a clear notation and a set of methodological guidelines for high-level Analysis & Design project documentation and prints it to a standard document structure.

## ***Ensemble by Cadre Technologies, Inc.***

### **Product Information:**

Version Number: 6.0  
 Date of Last Release: October 1994  
 Date of First Release: 1991  
 Number Sold: 4,000+  
 Single User Price: GSA Available

### **Contact Information:**

#### **Point of Contact:**

Name: Doug Trolan, Dir Fed Ops  
 Address: 6701 Democracy Plaza  
 Suite 710  
 Bethesda, MD 20817  
 Phone : 301-897-4101  
 Fax : 301-897-3106  
 E-mail : dtrolan@cadre.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	✓ Re-engineering
Requirements Analysis	✓ Low Level Design	✓ Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

All	✓ Engineering	✓ Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured                      Object-Oriented                      Behavior-Oriented

### **Configurations:**

Sparc(SunOS, Solaris), H-P PA RISC (HP-UX), and IBM RS6000 (AIX)

### **Description/Purpose:**

Ensemble is a suite of tightly integrated software tools for re-engineering legacy C code. The modules available as part of Ensemble include: System and Function Understanding for reverse engineering both embedded as well as host based C code into Structure Charts, module-specs, data dictionary entries and Control Flow Diagrams with code metrics; the Viewer for browsing and 3-D viewing of the relationships created by System and Function Understanding; Construction for regenerating the C source code from the Structure Charts, module specs and data dictionary entries; Test Verification provides code and branch coverage measurements; the Test Case Generator uses information in the data dictionary to generate black box test cases for each module and the Documentation modules assembles all the information in the CASE repository and formats it into reports. Ensemble stands alone or comes integrated with Teamwork.



## **ERWIN/ERX by LogicWorks**

### **Product Information:**

Version Number: 2.1  
Date of Last Release: May 1995  
Date of First Release: April 1990  
Number Sold: 16,000  
Single User Price: \$3,295 Single User  
\$1,996 GSA Single User

### **Contact Information:**

#### **Point of Contact:**

Name: David Donovan  
Address: 8245 Boone Blvd  
Suite 400  
Vienna, VA 22182  
Phone : 703-761-1166  
Fax : 703-761-1095  
E-mail :

### **Lifecycle Phases and Activities:**

- |                              |                     |                   |
|------------------------------|---------------------|-------------------|
| √ Requirements Specification | √ High Level Design | √ Re-engineering  |
| √ Requirements Analysis      | √ Low Level Design  | Environment       |
| √ Requirements Documentation | √ Database          | √ System Analysis |
| Requirements Tracing/Mgmt.   | √ Client-Server     | Other:            |

### **Intended Customers:**

- |            |                 |                  |
|------------|-----------------|------------------|
| All        | √ Engineering   | Embedded Systems |
| √ MIS      | √ Client-Server | Real-time        |
| √ Database | Other:          |                  |

### **Primary Methodology:**

- |              |                   |                   |
|--------------|-------------------|-------------------|
| √ Structured | √ Object-Oriented | Behavior-Oriented |
|--------------|-------------------|-------------------|

### **Configurations:**

MS-Windows, UNIX

### **Description/Purpose:**

ERWIN/ERX lets you use a point and click interface to create a graphical, Entity-Relationship (ER) model of the business rules governing your data. It is an IDEF1x tool which will generate high-quality database schema directly from your model and to reverse engineer existing schema into and a (ER) model. ERWIN/ERX and BPWIN cross validate business data and business process models.

## ***ES RE/Vision by Eden Systems Corp.***

### **Product Information:**

Version Number: 2.6  
 Date of Last Release: May 1995  
 Date of First Release: 1985  
 Number Sold: 150+ Organizations  
 Single User Price: Mainframe @ 55K  
 PC License for 25 users @ 25K  
 GSA Discount of 15%

### **Contact Information:**

#### **Point of Contact:**

Name: Brad Schweibold  
 Address: 9302 N. Meridian St.  
 Suite 350  
 Indianapolis, IN  
 46260-1820  
 Phone : 317-848-9600 ext 224  
 Fax : 317-483-2271  
 E-mail : custserv@iquest.net

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	✓ Re-engineering
Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	✓ System Analysis
Requirements Tracing/Mgmt.	Client-Server	
✓ Other: Portfolio Analysis, Quality Assurance and Control, Conversions (COBOL)		

### **Intended Customers:**

All	Engineering	Embedded Systems
✓ MIS	Client-Server	Real-time
Database	✓ Other: Application Development and Maintenance	

### **Primary Methodology:**

✓ Structured                      Object-Oriented                      Behavior-Oriented

### **Configurations:**

IBM(MVS), PC(DOS, Windows and OS/2)

### **Description/Purpose:**

ES RE/Vision is a standards checker and code improvement tool for COBOL and PL/1 applications. ES RE/Vision allows you to measure your programs against a user definable set of standards and rewrite your programs to meet your need for increased productivity, decreased risk, and COBOL code conversion.

## **EXPERT/CIO by P-Cube Corporation**

### **Product Information:**

Version Number: 3  
Date of Last Release: 1992  
Date of First Release: 1989  
Number Sold: ---  
Single User Price: ---

### **Contact Information:**

#### **Point of Contact:**

Name: Joseph Napoli  
Address: 572 E. Lambert Road  
Brea, CA 92621  
Phone Number: 714-990-3169  
Fax Number: 714-990-0838  
E-mail Address: ---

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
✓ Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	
✓ Other: Strategic/Tactical Planning, Simulation, Metrics		

### **Intended Customers:**

All	Engineering	Embedded Systems
✓ MIS	Client-Server	✓ Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured	Object-Oriented	Behavior-Oriented
--------------	-----------------	-------------------

### **Configurations:**

IBM XT/AT/PS2 (PC/MS DOS)

### **Description/Purpose:**

EXPERT/CIO provides a metrics-based framework for both strategic and tactical information systems planning and project selection. It directly links decisions on information technology to an organization's goals and objectives and can simulate the impact of new systems, the reengineering of existing systems, and the migration to new technologies. The cornerstone of the system is a well-defined methodology for surfacing and measuring information improvement opportunities and measuring the degree to which proposed information technology solutions will address those opportunities.

***firstcase by AGS Management Systems, Inc.***

**Product Information:**

Version Number: 3.4  
 Date of Last Release: May 1995  
 Date of First Release: 1992  
 Number Sold: 75  
 Single User Price: Concurrent Users:  
 Quantity Discounts

**Contact Information:**

**Point of Contact:**

Name: Valerie Palamountain  
 Address: 1012 W. Ninth Avenue  
 King of Prussia, PA 19406  
 Phone : 800-220-2471  
 610-265-1550  
 Fax : 610-265-1230  
 E-mail :

**Lifecycle Phases and Activities:**

- |                              |                     |                   |
|------------------------------|---------------------|-------------------|
| ✓ Requirements Specification | ✓ High Level Design | ✓ Re-engineering  |
| ✓ Requirements Analysis      | ✓ Low Level Design  | ✓ Environment     |
| ✓ Requirements Documentation | ✓ Database          | ✓ System Analysis |
| ✓ Requirements Tracing/Mgmt. | ✓ Client-Server     | Other:            |

**Intended Customers:**

- |          |                 |                  |
|----------|-----------------|------------------|
| ✓ All    | Engineering     | Embedded Systems |
| ✓ MIS    | ✓ Client-Server | ✓ Real-time      |
| Database | Other:          |                  |

**Primary Methodology:**

- |   |                 |                   |
|---|-----------------|-------------------|
| ✓ Structured  | Object-Oriented | Behavior-Oriented |
| ✓ Other: Information Engineering, Client/Server, Enterprise Architecture Planning |                 |                   |

**Configurations:**

Windows 3.1 or higher, OS/2 2.1 or higher, Windows NT

**Description/Purpose:**

firstcase is a process and project management system that provides a family of system development methodologies, estimating, project management and time reporting functions. firstcase operates on a LAN using a client/server architecture. A single, central database is used to store all methodologies, which can be customized by the client, all project and resource information. The central database design allows enterprise-wide process/project management.

## ***Foresight by Nu Thena Systems***

### **Product Information:**

Version Number: 4.0  
Date of Last Release: June 1995  
Date of First Release:  
Number Sold:  
Single User Price:

### **Contact Information:**

#### **Point of Contact:**

Name: Peter Stoupas  
Address: 1430 Spring Hill Rd.  
Suite 220  
McLean, VA 22102  
Phone : 703-356-5056  
Fax : 703-356-0498  
E-mail : info@nuthena.com

### **Lifecycle Phases and Activities:**

Requirements Specification	✓ High Level Design	Re-engineering
Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	✓ System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

All	✓ Engineering	✓ Embedded Systems
MIS	Client-Server	✓ Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured      ✓ Object-Oriented      ✓ Behavior-Oriented

### **Configurations:**

SUN (Unix) and HP (Unix)

### **Description/Purpose:**

Foresight is a tightly coupled suite of software tools for constructing and analyzing systems designs. Its advanced modeling and simulation capability allows the system developer to capture, test, analyze, and optimize any complex system. Foresight's robust modeling constructs capture all aspects of complex systems including performance, functionality, dynamic behavior, and information flow. With Foresight, you minimize development risks, times and costs.

## Foundation Design by Anderson Consulting Foundation

### Product Information:

Version Number: 7.0  
 Date of Last Release: May 1995  
 Date of First Release: 1980  
 Number Sold: 1,000+  
 Single User Price: \$6,500 list price

### Contact Information:

#### Point of Contact:

Name: Mary Ryan  
 Address: 69 W. Washington St.  
 Chicago, IL 60602  
 Phone : 800-458-8851  
 Fax :  
 E-mail :

### Lifecycle Phases and Activities:

✓ Requirements Specification	✓ High Level Design	✓ Re-engineering
✓ Requirements Analysis	✓ Low Level Design	✓ Environment
✓ Requirements Documentation	✓ Database	✓ System Analysis
✓ Requirements Tracing/Mgmt.	✓ Client-Server	Other:

### Intended Customers:

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### Primary Methodology:

✓ Structured	Object-Oriented	Behavior-Oriented
--------------	-----------------	-------------------

### Configurations:

OS/2

### Description/Purpose:

Design/1 supplies a full set of tools to manage risk and forms a complete application development and documentation solution. These tools

- 1) Provide package-knowledge through the reuse of deliverables across multiple application and development stages via LAN-based repository.
- 2) Increase developer productivity through robust tools and standard templates to get you started quickly.
- 3) Integrate with a robust methodology to enable developers to easily identify what to do and how to complete their assigned deliverables.

## ***Foundation Vista by Menlo Business Systems***

### **Product Information:**

Version Number: 4.8  
 Date of Last Release: 1994  
 Date of First Release: 1987  
 Number Sold: 300+  
 Single User Price: \$2,900 (Qty 1)  
 Volume Discounts

### **Contact Information:**

#### **Point of Contact:**

Name: Ronald Snow  
 Address: 201 Main Street  
 Los Altos, CA 94022  
 Phone : 415-948-7920  
 Fax : 415-949-6655  
 E-mail : menlomail@eWorld.com

### **Lifecycle Phases and Activities:**

√ Requirements Specification	√ High Level Design	√ Re-engineering
√ Requirements Analysis	√ Low Level Design	Environment
√ Requirements Documentation	√ Database	System Analysis
Requirements Tracing/Mgmt.	√ Client-Server	Other:

### **Intended Customers:**

√ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

√ Structured	Object-Oriented	Behavior-Oriented
--------------	-----------------	-------------------

### **Configurations:**

Macintosh and PowerPC, System 6, System 7+

### **Description/Purpose:**

FOUNDATION VISTA automates the documentation and diagramming process. Designers produce high-quality system documentation, business process flows, data flow diagrams, database design, (NonStop SQL, Enscribe, ORACLE, DB2, SYBASE... others) program structure definitions, forms and screen designs. It also validates output against accepted editor rules throughout the stages of development. FOUNDATION VISTA takes full advantage of the ease of use, power, and flexibility of the Macintosh advanced workstation. FOUNDATION VISTA supports Yourdon, Gane-Sarson, Jackson and other structured design methodologies. All modules are interfaced through the central, multi-user data dictionary. Dictionary information can be exported and used as the source for software generation. Dictionary access supports both the single workstation and a multi-user LAN environment.

***IE Advantage by Information Engineering Systems Corp.***

**Product Information:**

Version Number: 6.1.4  
Date of Last Release: May 1994  
Date of First Release: March 1990  
Number Sold: 1,000+  
Single User Price: 1-4 = \$8,750  
5+ = \$7,500

**Contact Information:**

**Point of Contact:**

Name: Fred Krause  
Address: 210 N. Union Street  
Alexandria, VA 22314  
Phone : 703-739-2242  
Fax : 703-739-0074  
E-mail :

**Lifecycle Phases and Activities:**

√ Requirements Specification	√ High Level Design	√ Re-engineering
√ Requirements Analysis	√ Low Level Design	√ Environment
√ Requirements Documentation	√ Database	√ System Analysis
√ Requirements Tracing/Mgmt.	√ Client-Server	
√ Other: Business Process Modeling		

**Intended Customers:**

√ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

**Primary Methodology:**

√ Structured      √ Object-Oriented      Behavior-Oriented

**Configurations:**

386/486 PC, Windows/Windows NT/Novell

**Description/Purpose:**

Automates the definition and refinement of business plans, IE and IDEF1x modeling, business and systems process modeling (Version 7.0 in beta now will include activity-based costing and IDEF0 activity modeling), object-oriented database and application design, plus interface to repository and 4GL, 5GL products, plus free utilities for systems reverse engineering, plus extensive report and document generation.



## ***Infomodeler by Asymetrix Corp.***

### **Product Information:**

Version Number: 1.5  
Date of Last Release: November 1994  
Date of First Release: February 1994  
Number Sold: 10,000  
Single User Price: \$395 - \$1,995

### **Contact Information:**

#### **Point of Contact:**

Name: Lance Wieland  
Address: 110 100th Ave NE  
  
Phone : 800-471-5184 x2465  
Fax : 206-637-1504  
E-mail : lancew@asymetrix.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	Environment
Requirements Documentation	✓ Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

All	Engineering	Embedded Systems
MIS	✓ Client-Server	Real-time
✓ Database	Other:	

### **Primary Methodology:**

Structured	Object-Oriented	Behavior-Oriented
✓ Other		

### **Configurations:**

Windows based

### **Description/Purpose:**

Data Modeling Tool - See product Literature.

***iRAT by introspect Technologies, Inc.***

**Product Information:**

Version Number: 1.3  
 Date of Last Release: February 1995  
 Date of First Release: September 1994  
 Number Sold: 12  
 Single User Price: \$3,595 Single Seat

**Contact Information:**

**Point of Contact:**

Name: Mary Burns/Cynthia Mavros  
 Address: 1765 S. 8th St.  
 Suite T8  
 Colorado Springs, CO 80906  
 Phone : 719-634-5744  
 Fax : 719-634-1163  
 E-mail : intros@usa.net

**Lifecycle Phases and Activities:**

Requirements Specification	✓ High Level Design	✓ Re-engineering
✓ Requirements Analysis	Low Level Design	✓ Environment
✓ Requirements Documentation	Database	✓ System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

**Intended Customers:**

All	✓ Engineering	✓ Embedded Systems
MIS	Client-Server	✓ Real-time
Database	✓ Other: System Analysis, Requirements, Development	

**Primary Methodology:**

✓ Structured      ✓ Object-Oriented      ✓ Behavior-Oriented

**Configurations:**

PC 286 or better, 2 meg RAM; 5 meg of hard disk space  
 MS DOS 3.0 or later; MS Windows 3.1

**Description/Purpose:**

iRAT is an advance software engineering tool used to analyze runtime performance of time-sensitive systems. Based on fixed-priority scheduling theory and Rate Monotonic Analysis (RMA), iRAT can be applied to any application with time-critical requirements (e.g., avionics, robotics, manufacturing, biomedical, communications). iRAT enhances the user's ability to understand and predict the time behavior of a system. iRAT can be used to support development decisions during requirements, design, integration, or upgrade phases. From identifying potential timing and scheduling risks to troubleshooting existing performance problems -- iRAT's unique capabilities allow analysis of real-time systems before, during and after implementation. iRAT helps guarantee that timing requirements are always met!

## ***LINC Environment by Unisys Corp.***

### **Product Information:**

Version Number: 15.4  
Date of Last Release: 1995  
Date of First Release: 1982  
Number Sold: 3500  
Single User Price: 400 a user for  
deployment  
Named User - 12,500 for Development  
Concurrent User - 15,000 for Development  
Discounts available

### **Contact Information:**

#### **Point of Contact:**

Name: Gary Olsen  
Address: PO Box 500  
Blue Bell, PA 19424  
Phone : 215-986-3278  
Fax : 215-986-6646  
E-mail : linc@p03.bb.unisys.com

### **Lifecycle Phases and Activities:**

√ Requirements Specification	√ High Level Design	Re-engineering
Requirements Analysis	√ Low Level Design	√ Environment
Requirements Documentation	√ Database	System Analysis
Requirements Tracing/Mgmt.	√ Client-Server	Other:

### **Intended Customers:**

All	√ Engineering	Embedded Systems
√ MIS	√ Client-Server	Real-time
√ Database	Other:	

### **Primary Methodology:**

Structured      √ Object-Oriented      Behavior-Oriented

### **Configurations:**

Unisys A series, 2200 series and U series.  
Sun, Sequent, RS6000 - (SVR4, PTX and AIX)

### **Description/Purpose:**

The LINC Environment is business-oriented software for rapid solution development of high-volume, mission-critical transaction processing applications. It facilitates design, development, generation, and maintenance, and has the facility to fast-track software changes that might otherwise languish in the software developer's to-do list. LINC is used in three activities; development, system generation, and deployment of the generated system. LINC Development provides environments to handle analysis, design, specification, unit test, debugging, and prototyping. LINC Generate automates coding and integration of the complete system ready for deployment.

## MacAnalyst & Mac Designer by Excel Software

### Product Information:

Version Number: 4.3  
 Date of Last Release: January 1995  
 Date of First Release: 1986  
 Number Sold: 1000+  
 Single User Price: \$2,995/copy  
 Other product options and site licensing available

### Contact Information:

#### Point of Contact:

Name: Harold Halbleib  
 Address: PO Box 1414  
 Marshalltown, IA 50518  
 Phone : 515-752-5359  
 Fax : 515-752-2435  
 E-mail : casetools@aol.com

### Lifecycle Phases and Activities:

- |                              |                     |                   |
|------------------------------|---------------------|-------------------|
| ✓ Requirements Specification | ✓ High Level Design | ✓ Re-engineering  |
| ✓ Requirements Analysis      | ✓ Low Level Design  | ✓ Environment     |
| ✓ Requirements Documentation | ✓ Database          | ✓ System Analysis |
| ✓ Requirements Tracing/Mgmt. | ✓ Client-Server     | Other:            |

### Intended Customers:

- |          |               |                  |
|----------|---------------|------------------|
| ✓ All    | Engineering   | Embedded Systems |
| MIS      | Client-Server | Real-time        |
| Database | Other:        |                  |

### Primary Methodology:

- ✓ Structured      ✓ Object-Oriented      ✓ Behavior-Oriented

### Configurations:

Macintosh (System 7 or A/UX 3), Sun & HP UNIX (Requires Apple's MAE software)

### Description/Purpose:

MacAnalyst and MacDesigner support a variety of computer aided software engineering methods for personal computer, mainframe, or embedded software projects. Structured analysis and design methods include Yourdon/DeMarco and Gane/Sarson. Real-time extensions include Mealy, Moore and Hatley/Pirbhai state and control flow models. Chen ER diagrams, screen prototypes, and the Information Engineering method support database design. Object-oriented analysis and design methods include Shlaer/Mellor, Coad/Yourdon, Booch, and Rumbaugh. Code generation includes, C, C++, Pascal, Object Pascal, Basic, FORTRAN, and SQL. All products are compatible with Macintosh computers running System 7, A/UX 3, or later and some UNIX computers using Apple's MAE software.

While MacAnalyst and MacDesigner are generally considered to be language independent tools, the Translator utility is used to reengineer existing source code for C, C++, Pascal, Object Pascal, FORTRAN and Basic. Together with MacAnalyst and MacDesigner, it can automatically generate structure charts, class diagrams, populate the data dictionary and provide double-click access from diagrams to code.

## ***Magec RAD System by Magec Software***

### **Product Information:**

Version Number: 3.0  
Date of Last Release: March 1995  
Date of First Release: June 1981  
Number Sold: 3,200+  
Single User Price: \$2,000 (PC) to  
\$125,000 (MVS)

### **Contact Information:**

#### **Point of Contact:**

Name: Vic Lee  
Address: 1603 LBJ Freeway  
Suite 880  
Dallas, TX 75234  
Phone : 800-336-2432  
Fax : 214-488-3030  
E-mail :

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	√ Re-engineering
Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	√ Client-Server	Other:

### **Intended Customers:**

√ All	Engineering	Embedded Systems
√ MIS	√ Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

Structured	Object-Oriented	Behavior-Oriented
√ WYSIWYG Prototyping		

### **Configurations:**

IBM & compatible mainframe - MVS/DOS/VSE  
IBM & compatible Midrange - RS6000 AIX  
IBM & compatible PC/LAN - DOS, Windows, NT, OS/2

### **Description/Purpose:**

Automated design and development of COBOL applications that are transparently portable across platforms, client/server compatible, platform and database independent, and (optionally) contain GUI capabilities.

**McCabe Object-Oriented Tool™ by McCabe and Associates, Inc.**

**Product Information:**

Version Number: V5  
 Date of Last Release: April 1995  
 Date of First Release: December 1993  
 Number Sold:  
 Single User Price:

**Contact Information:**

**Point of Contact:**

Name: Barbara Fiato  
 Address: 5501 Twin Knolls Rd. #111  
 Columbia, MD 21045  
 Phone : 800-638-6316  
 Fax : 410-995-1528  
 E-mail : Mike@McCabe.com

**Lifecycle Phases and Activities:**

✓ Requirements Specification	✓ High Level Design	✓ Re-engineering
✓ Requirements Analysis	✓ Low Level Design	✓ Environment
✓ Requirements Documentation	Database	System Analysis
✓ Requirements Tracing/Mgmt.	Client-Server	
✓ Other:		

**Intended Customers:**

All	✓ Engineering	✓ Embedded Systems
MIS	✓ Client-Server	✓ Real-time
Database	✓ Other: Developers	

**Primary Methodology:**

✓ Structured                      Object-Oriented                      ✓ Behavior-Oriented

**Configurations:**

Operate on the following platforms: SUN4, IBM RS6000, Data General, DEC Alpha, DEC VMS, DEC RISC/Ultrix, Apollo/Domain, HP9000/300, 400, 700, Silicon Graphics and IBM PC.

**Description/Purpose:**

The McCabe Object-Oriented Tool™ provides clear visualization of OO application architecture. It displays an OO system from the perspective of a single object or from that of a group of objects. It also supports the three concepts of inheritance, encapsulation and polymorphism. The Tool provides system metrics including both traditional and new McCabe OO metrics and OO industry standard metrics. It also includes the concept of SAFE/UNSAFE to designate tested and untested classes and fully tests the integration of C++ classes while keeping the testing effort manageable.

## ***MethodMaker TDK by Mark V Systems***

### **Product Information:**

Version Number: 3.2  
Date of Last Release: June 1995  
Date of First Release: June 1992  
Number Sold: 250  
Single User Price: \$10,000

### **Contact Information:**

#### **Point of Contact:**

Name: Mo Bjornestad  
Address: 16400 Ventura Blvd #300  
Encino, CA 91436-2123  
Phone : 818-995-7671  
Fax : 818-995-4267  
E-mail : mob@markv.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	
√ Other: Meta Tool		

### **Intended Customers:**

√ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

Structured      √ Object-Oriented      Behavior-Oriented

### **Configurations:**

Microsoft Windows (3.1, NT, 95)  
most UNIX platforms (IBM, HP, SUN, SGI, DEC)

### **Description/Purpose:**

## **ObjectMaker® by Mark V Systems**

### **Product Information:**

Version Number: 4.0  
 Date of Last Release: June 1995  
 Date of First Release: September 1985  
 Number Sold: 4000  
 Single User Price: \$810  
                               \$1,350  
                               \$3,350

### **Contact Information:**

#### **Point of Contact:**

Name: Mo Bjornestad  
 Address: 16400 Ventura Blvd #300  
           Encino, CA 91436-2123  
 Phone : 818-995-7671  
 Fax : 818-995-4267  
 E-mail : mob@markv.com

### **Lifecycle Phases and Activities:**

✓ Requirements Specification ✓ Requirements Analysis ✓ Requirements Documentation Requirements Tracing/Mgmt.	✓ High Level Design ✓ Low Level Design ✓ Database ✓ Client-Server	✓ Re-engineering ✓ Environment ✓ System Analysis Other:
---	--	--

### **Intended Customers:**

✓ All MIS Database	Engineering Client-Server Other:	Embedded Systems Real-time
--------------------------	--	-------------------------------

### **Primary Methodology:**

✓ Structured                      ✓ Object-Oriented                      ✓ Behavior-Oriented

### **Configurations:**

Microsoft Windows (3.1, NT, 95)  
 most UNIX platforms (IBM, HP, SUN, SGI, DEC)

### **Description/Purpose:**



## **ObjectMaker TDK by Mark V Systems**

### **Product Information:**

Version Number: 4.0  
Date of Last Release: June 1995  
Date of First Release: March 1980  
Number Sold: 200  
Single User Price: \$24,500

### **Contact Information:**

#### **Point of Contact:**

Name: Mo Bjornestad  
Address: 16400 Ventura Blvd #300  
Encino, CA 91436-2123  
Phone : 818-995-7671  
Fax : 818-995-4267  
E-mail : mob@markv.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	
√ Other: Meta Tool		

### **Intended Customers:**

√ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

Structured      √ Object-Oriented      Behavior-Oriented

### **Configurations:**

Microsoft Windows (3.1, NT, 95)  
most UNIX platforms (IBM, HP, SUN, SGI, DEC)

### **Description/Purpose:**

## ***ObjectTeam for Shlaer-Mellor by Cadre Technologies, Inc.***

### **Product Information:**

Version Number: 6.0  
 Date of Last Release: September 1994  
 Date of First Release: 1988  
 Number Sold: 2,200+  
 Single User Price: GSA Available

### **Contact Information:**

#### **Point of Contact:**

Name: Doug Trolan, Dir Fed Ops  
 Address: 6701 Democracy Plaza  
 Suite 710  
 Bethesda, MD 20817  
 Phone : 301-897-4101  
 Fax : 301-897-3106  
 E-mail : dtrolan@cadre.com

### **Lifecycle Phases and Activities:**

√ Requirements Specification	√ High Level Design	Re-engineering
√ Requirements Analysis	√ Low Level Design	√ Environment
√ Requirements Documentation	Database	√ System Analysis
√ Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

All	√ Engineering	√ Embedded Systems
MIS	Client-Server	√ Real-time
Database	Other:	

### **Primary Methodology:**

Structured      √ Object-Oriented      √ Behavior-Oriented

### **Configurations:**

Sparc(SunOS, Solaris), H-P PA RISC (HP-UX), IBM RS6000 (AIX), DEC ALPHA (OSF/1), DEC (OPEN VMS), SGI(IRIX), DEC(ULTRIX), Intel (OS/2), Intel (Solaris).

### **Description/Purpose:**

ObjectTeam for Shlaer-Mellor provides comprehensive automation support for the Shlaer-Mellor Object-Oriented Analysis and Recursive Design methodology. The tool supports system, domain, sub-system and object-oriented analysis level diagrams with comprehensive consistency and completeness checking and reporting. The tool suite automatically generates derived work products such as the Subsystem and Object Communication Diagrams. The OOA/SIM product (Release 6.1 in June 1995) provides the ability to execute the OO Analysis models. The OOD product supports the drawing of OODLE notation for System Architecture design. Early support for Recursive Design (code generation) is included with an example for C++ code generation for the Object and State Models.

**ObjectTeam/ProDev by Cadre Technologies, Inc.**

**Product Information:**

Version Number: 1.0  
Date of Last Release: March 1995  
Date of First Release: March 1995  
Number Sold: 10  
Single User Price: Discounts Available

**Contact Information:**

**Point of Contact:**

Name: Doug Trolan, Dir Fed Ops  
Address: 6701 Democracy Plaza  
Suite 710  
Bethesda, MD 20817  
Phone : 301-897-4101  
Fax : 301-897-3106  
E-mail : dtrolan@cadre.com

**Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	✓ Re-engineering
Requirements Analysis	✓ Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

**Intended Customers:**

All	✓ Engineering	Embedded Systems
✓ MIS	Client-Server	Real-time
Database	Other:	

**Primary Methodology:**

✓ Structured      ✓ Object-Oriented      Behavior-Oriented

**Configurations:**

SPARC (Solaris), H-P PA RISC (HP-UX) (Planned), IBM RS6000(AIX)(Planned)

**Description/Purpose:**

ObjectTeam/ProDev is Cadre's highly visual programming environment for C/C++ that includes an integrated static analyzer, debugger, C++ browsers, make analyzer, tester (planned) and performance analyzer (planned). ObjectTeam/ProDev can handle very large C/C++ source files (100 mbytes+) with minimal degradation of user response time and without making changes to the make file. The tool offers easy integration with popular source code configuration management tools.

## ***ObjectTeam for OMT by Cadre Technologies, Inc.***

### **Product Information:**

Version Number: 3.1  
 Date of Last Release: March 1995  
 Date of First Release: March 1994  
 Number Sold: 110+  
 Single User Price: Discounts Available

### **Contact Information:**

#### **Point of Contact:**

Name: Doug Trolan, Dir Fed Ops  
 Address: 6701 Democracy Plaza  
 Suite 710  
 Bethesda, MD 20817  
 Phone : 301-897-4101  
 Fax : 301-897-3106  
 E-mail : dtrolan@cadre.com

### **Lifecycle Phases and Activities:**

√ Requirements Specification	√ High Level Design	Re-engineering
√ Requirements Analysis	√ Low Level Design	Environment
√ Requirements Documentation	√ Database	System Analysis
√ Requirements Tracing/Mgmt.	√ Client-Server	Other:

### **Intended Customers:**

√ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

Structured      √ Object-Oriented      Behavior-Oriented

### **Configurations:**

Sparc(SunOS(planned), Solaris), H-P PA RISC (HP-UX), IBM RS6000 (AIX), DEC ALPHA (OSF/1), SGI(IRIX)(planned), Windows 3.1 (client planned), Windows 95 (Client planned).

### **Description/Purpose:**

ObjectTeam for OMT is a multi-user CASE capability for automating the Object Modeling Technique with extensions. The toolset supports four phases of development by iteration and elaboration including: Analysis, System Design, Object Design and Implementation. OMT diagrams supported include the Object Diagram, Harel State Diagrams, Process Diagrams and Event-Trace Diagrams. Extensions to OMT include the Class Communication Diagram and the Message Generalization Diagram. The toolset supports the generation of C++ class definitions, Ada (planned) and New Era OO 4GL from Informix. Other code generators are planned. Integration with RTM, DOORS and DocEXPRESS are planned.

## **Obsydian by Synon, Inc.**

### **Product Information:**

Version Number: 1.0.3  
Date of Last Release: May 1995  
Date of First Release: October 1994  
Number Sold: 250  
Single User Price: Single User \$8,750  
Quantity Discounts Available

### **Contact Information:**

#### **Point of Contact:**

Name: Byron Wilkes  
Address: 1001 Fourth Ave, #3200  
Seattle, WA 98154  
Phone : 206-223-1404  
Fax : 206-382-9648  
E-mail :

### **Lifecycle Phases and Activities:**

✓ Requirements Specification	✓ High Level Design	✓ Re-engineering
✓ Requirements Analysis	✓ Low Level Design	✓ Environment
✓ Requirements Documentation	✓ Database	System Analysis
✓ Requirements Tracing/Mgmt.	✓ Client-Server	Other:

### **Intended Customers:**

✓ All	✓ Engineering	Embedded Systems
✓ MIS	✓ Client-Server	Real-time
✓ Database	Other:	

### **Primary Methodology:**

Structured                      Object-Oriented                      Behavior-Oriented

### **Configurations:**

DOS/Windows, Windows 95, Eventually Windows/NT

### **Description/Purpose:**

Obsydian is a desktop application toolset that combines Information Engineering (High-Level Abstraction) and Data Modeling with Object Orientation Concepts/Features of Inheritance, Reuse, etc. The product covers the entire life cycle from Analysis, Design, Prototyping, Construction, and Maintenance. It provides Version Control/Change Management as well as one integrated product. Deployment is a client/server (multi-tiered) strategy. Servers include AS400, ODBC compliant, and UNIX platforms.

## ***PACE by Wang Laboratories, Inc.***

<b>Product Information:</b>  Version Number: 1.3 Date of Last Release: February 1995 Date of First Release: February 1993 Number Sold: -- Single User Price: (Client) \$295 - \$995 (Server) \$3500 - \$128,500	<b>Contact Information:</b>  <b>Point of Contact:</b> Name: Judy Cole Address: Wang Labs. M/S OIS-430 600 Technology Park Dr. Billerica, MA Phone : 508-967-2199 Fax : 508-967-9736 E-mail : judy.cole@mailoff.wang.com
--	--

<b>Lifecycle Phases and Activities:</b>		
Requirements Specification Requirements Analysis Requirements Documentation Requirements Tracing/Mgmt.	High Level Design Low Level Design Database Client-Server	Re-engineering Environment System Analysis Other:

<b>Intended Customers:</b>		
All ✓ MIS ✓ Database	Engineering ✓ Client-Server Other:	Embedded Systems Real-time

<b>Primary Methodology:</b>		
✓ Structured	Object-Oriented	Behavior-Oriented

<b>Configurations:</b>
Client: Intel PC 386 or higher (MS-DOS, Microsoft Windows) Server: IBM RS/6000(AIX), WANG VS(VS), HP9000 (HP-UX)

<b>Description/Purpose:</b>
PACE is a set of application development and information management tools built upon a rules-based relational database. PACE provides a complete solution for the rapid development and implementation of full-featured client/server database applications for the windows operating system environment.

STSC RED Product Sheet Version 2.0c

## ***Paradigm Plus by ProtoSoft***

### **Product Information:**

Version Number: 2.01a  
Date of Last Release: April 95  
Date of First Release: August 86  
Number Sold: \$5,000  
Single User Price: \$3,995 and up

### **Contact Information:**

#### **Point of Contact:**

Name: Kevin Shea  
Address: 3900 University Drive #210  
Fairfax, VA 22030  
Phone : 703-359-3755  
Fax : 703-359-3753  
E-mail : shea@protosoft.com

### **Lifecycle Phases and Activities:**

- |                              |                     |                   |
|------------------------------|---------------------|-------------------|
| √ Requirements Specification | √ High Level Design | √ Re-engineering  |
| √ Requirements Analysis      | √ Low Level Design  | √ Environment     |
| √ Requirements Documentation | √ Database          | √ System Analysis |
| Requirements Tracing/Mgmt.   | √ Client-Server     | Other:            |

### **Intended Customers:**

- |          |               |                  |
|----------|---------------|------------------|
| √ All    | Engineering   | Embedded Systems |
| MIS      | Client-Server | Real-time        |
| Database | Other:        |                  |

### **Primary Methodology:**

- Structured      √ Object-Oriented      Behavior-Oriented

### **Configurations:**

PC - MS Windows, Windows NT, Windows 95  
UNIX - Sun OS, Solaris, Alpha OSF/1, Silicon Graphics, HP-UX, AIX, AT&T  
Network - LAN Manager, TCP/IP, Novell, NetBIOS

### **Description/Purpose:**

Paradigm Plus is a powerful application development tool that supports Enterprise Component Modeling (ECM), code generation, and reverse engineering. Using ECM, companies can identify business requirements, model reusable application components and manage systems over the long term. Paradigm Plus makes ECM possible through its support for all leading object-oriented methodologies, incorporation of a powerful object repository for large teams of concurrent users and automatic synchronization of models, source code, and documentation. Using methods such as OMT, Fusion, Martin/Odell OOIE and others extended to support Jacobson's Use Case modeling, Paradigm Plus automates all leading object-oriented methods in a single solution. Automatic generation of C, C++, Smalltalk, Ada, ODBMS and RDBMS schema definitions increases productivity. Built-in consistency checks ensure quality and reduce maintenance cost, while a powerful script language allows custom reporting, checking and code generation. Reverse engineering utilities preserve your investment in legacy software. On-line hypertext help, thorough documentation and hands-on tool training give your team a quick start. Paradigm Plus is available on most popular UNIX and PC platforms.

STSC RED Product Sheet Version 2.0c

## **ProcessMaker® by Mark V Systems**

### **Product Information:**

Version Number: 4.0  
 Date of Last Release: June 1995  
 Date of First Release: June 1992  
 Number Sold: 500  
 Single User Price: \$810  
 \$1,350  
 \$3,350

### **Contact Information:**

#### **Point of Contact:**

Name: Mo Bjornestad  
 Address: 16400 Ventura Blvd #300  
 Encino, CA 91436-2123  
 Phone : 818-995-7671  
 Fax : 818-995-4267  
 E-mail : mob@markv.com

### **Lifecycle Phases and Activities:**

√ Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	
√ Other: Process & Workflow Modeling		

### **Intended Customers:**

√ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

√ Structured      √ Object-Oriented      √ Behavior-Oriented

### **Configurations:**

Microsoft Windows (3.1, NT, 95)  
 most UNIX platforms (IBM, HP, SUN, SGI, DEC)

### **Description/Purpose:**



## ***ProMod-AutoDesign by G & E Systems***

### **Product Information:**

Version Number: 3.1  
Date of Last Release: January 95  
Date of First Release: July 80  
Number Sold: 30,000  
Single User Price: \$2,400  
Quantity discounts available

### **Contact Information:**

#### **Point of Contact:**

Name: Bryan Cooper  
Address: 250 Edge Hill Rd.  
Sharon, MA 02067  
Phone : 617-784-1007  
Fax : 617-784-1737  
E-mail : bkocher@aol.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	✓ Low Level Design	Environment
Requirements Documentation	Database	System Analysis
✓ Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured      ✓ Object-Oriented      ✓ Behavior-Oriented

### **Configurations:**

PC (Windows, OS/2 and NT), VAX (VMS), DEC (Ultrix), SUN Sparc (Solaris), HP 9000 (HP-UX), and IBM RS6000 (AIX).

### **Description/Purpose:**

ProMod-AutoDesign automates generation of Modular Designs from Structured Analysis and automatically maintains all linkages between Modular Design and Structured Analysis objects. Analysts and Programmers can directly access design documentation to ensure proper design. Also includes Task Designer for concurrent process design.

## ***ProMod-GUI by G & E Systems***

### **Product Information:**

Version Number: 3.1  
 Date of Last Release: January 95  
 Date of First Release: July 80  
 Number Sold: --  
 Single User Price: \$5,395

### **Contact Information:**

#### **Point of Contact:**

Name: Bryan Cooper  
 Address: 250 Edge Hill Rd.  
 Sharon, MA 02067  
 Phone : 617-784-1007  
 Fax : 617-784-1737  
 E-mail : bkocher@aol.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	✓ Re-engineering
Requirements Analysis	✓ Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	✓ Other:

### **Intended Customers:**

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured      ✓ Object-Oriented      Behavior-Oriented

### **Configurations:**

PC (Windows, OS/2 and NT), VAX (VMS), DEC (Ultrix), SUN Sparc (Solaris), HP 9000 (HP-UX), and IBM RS6000 (AIX).

### **Description/Purpose:**

ProMod-GUI allows design and build of Graphical User Interfaces as part of the design process. It includes: graphical screen painter, object oriented dialog script language, portable user interface for highest function on each platform, interfaces to C, C++, and COBOL. Combine with reverse engineering to modernize legacy applications.

## **ProMod-OCM by G & E Systems**

### **Product Information:**

Version Number: 3.1  
Date of Last Release: January 95  
Date of First Release:  
Number Sold: 5,000  
Single User Price: \$5,800  
Only one needed per LAN

### **Contact Information:**

#### **Point of Contact:**

Name: Bryan Cooper  
Address: 250 Edge Hill Rd.  
Sharon, MA 02067  
Phone : 617-784-1007  
Fax : 617-784-1737  
E-mail : bkocher@aol.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	✓ Other:

### **Intended Customers:**

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured      ✓ Object-Oriented      Behavior-Oriented

### **Configurations:**

DEC (Ultrix), SUN Sparc (Solaris), HP 9000 (HP-UX), and IBM RS6000 (AIX).

### **Description/Purpose:**

ProMod-OCM is an Object Oriented database manager for shared, multi-user client/server oriented system design teams. OCM allows multiple, concurrent users to work on the same design project and resolves any design conflicts as objects are stored. Includes API for all ProMod components.

## ***ProMod-Plus by G & E Systems***

### **Product Information:**

Version Number: 3.1  
 Date of Last Release: January 95  
 Date of First Release: July 1980  
 Number Sold: 30,000  
 Single User Price: \$1,295 Base  
 Quantity Discounts available

### **Contact Information:**

#### **Point of Contact:**

Name: Bryan Cooper  
 Address: 250 Edge Hill Rd.  
 Sharon, MA 02067  
 Phone : 617-784-1007  
 Fax : 617-784-1737  
 E-mail : bkocher@aol.com

### **Lifecycle Phases and Activities:**

- |                              |                     |                   |
|------------------------------|---------------------|-------------------|
| ✓ Requirements Specification | ✓ High Level Design | ✓ Re-engineering  |
| ✓ Requirements Analysis      | ✓ Low Level Design  | ✓ Environment     |
| ✓ Requirements Documentation | ✓ Database          | ✓ System Analysis |
| ✓ Requirements Tracing/Mgmt. | ✓ Client-Server     | Other:            |

### **Intended Customers:**

- |          |               |                  |
|----------|---------------|------------------|
| ✓ All    | Engineering   | Embedded Systems |
| MIS      | Client-Server | Real-time        |
| Database | Other:        |                  |

### **Primary Methodology:**

- ✓ Structured      ✓ Object-Oriented      ✓ Behavior-Oriented

### **Configurations:**

PC (Windows, OS/2 and NT), VAX (VMS), DEC (Ultrix), SUN Sparc (Solaris), HP 9000 (HP-UX), and IBM RS6000 (AIX).

### **Description/Purpose:**

ProMod™ is a multi-windowed graphical development environment, supporting most popular methodologies for structure analysis, information modeling, modular design and real-time design. Data dictionary tracks all objects. Runs on Ultrix, VMS, Windows, O/S2, AIX, HP-UX, Solaris, etc. Perfect for open systems environment.

## ***ProMod-PROFACE by G & E Systems***

### **Product Information:**

Version Number: 3.1  
Date of Last Release: January 95  
Date of First Release: July 80  
Number Sold: --  
Single User Price: \$2,400

### **Contact Information:**

#### **Point of Contact:**

Name: Bryan Cooper  
Address: 250 Edge Hill Rd.  
Sharon, MA 02067  
Phone : 617-784-1007  
Fax : 617-784-1737  
E-mail : bkocher@aol.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	√ Re-engineering
Requirements Analysis	√ Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	√ Other:

### **Intended Customers:**

√ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

√ Structured      √ Object-Oriented      Behavior-Oriented

### **Configurations:**

DEC (Ulrix), SUN Sparc (Solaris), HP 9000 (HP-UX), and IBM RS6000 (AIX).

### **Description/Purpose:**

ProMod-ProFace generates UNIFACE (4GL) database schemas and source code directly from Modular Design. Using ProMod to design and ProFace to generate the database and code, ensures that the documentation always perfectly represents the implemented system. ProFace also performs reverse engineering.

## ***ProMod-ProSQL by G & E Systems***

### **Product Information:**

Version Number: 1.1  
 Date of Last Release: January 95  
 Date of First Release: January 95  
 Number Sold: --  
 Single User Price: \$5,395 Base

### **Contact Information:**

#### **Point of Contact:**

Name: Bryan Cooper  
 Address: 250 Edge Hill Rd.  
 Sharon, MA 02067  
 Phone : 617-784-1007  
 Fax : 617-784-1737  
 E-mail : bkocher@aol.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	✓ Low Level Design	Environment
Requirements Documentation	✓ Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured      ✓ Object-Oriented      Behavior-Oriented

### **Configurations:**

PC (Windows, OS/2 and NT), VAX (VMS), DEC (Ultrix), SUN Sparc (Solaris), HP 9000 (HP-UX), and IBM RS6000 (AIX).

### **Description/Purpose:**

ProMod-ProSQL is a SQL database designer that generates DDL (data definition language) for Oracle 6 & 7 and Informix. Databases are defined through easy to use dialogs/forms.

## ***ProMod-SourcePilot by G & E Systems***

### **Product Information:**

Version Number: 3.1  
Date of Last Release: January 95  
Date of First Release: --  
Number Sold: 2,000  
Single User Price: \$3,400

### **Contact Information:**

#### **Point of Contact:**

Name: Bryan Cooper  
Address: 250 Edge Hill Rd.  
Sharon, MA 02067  
Phone : 617-784-1007  
Fax : 617-784-1737  
E-mail : bkocher@aol.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	✓ Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	✓ Other:

### **Intended Customers:**

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured      ✓ Object-Oriented      Behavior-Oriented

### **Configurations:**

DEC (Ultrix), SUN Sparc (Solaris), HP 9000 (HP-UX), and IBM RS6000 (AIX).

### **Description/Purpose:**

ProMod-SourcePilot generates C or FORTRAN source code directly from a Modular Design. Using ProMod CASE to design systems and SourcePilot to generate the code ensures that the documentation for a system always perfectly represents the code. SourcePilot also performs reverse engineering - create documentation from old code.

## ***Provision Workbench by Performa Corp.***

### **Product Information:**

Version Number: V1-R1  
 Date of Last Release: --  
 Date of First Release: Planned for July 95  
 Number Sold: --  
 Single User Price: Intro Price \$911

### **Contact Information:**

#### **Point of Contact:**

Name: Hugh Mensch  
 Address: 17515 West 9-Mile Rd.  
 Southfield, MI 48075  
 Phone : 810-443-0506  
 Fax : 810-443-5495  
 E-mail :

### **Lifecycle Phases and Activities:**

√ Requirements Specification	High Level Design	Re-engineering
√ Requirements Analysis	Low Level Design	√ Environment
√ Requirements Documentation	√ Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	
√ Other: Business Process Re-engineering		

### **Intended Customers:**

All	Engineering	Embedded Systems
√ MIS	√ Client-Server	Real-time
Database	√ Other: BPR Practioners	

### **Primary Methodology:**

Structured      √ Object-Oriented      Behavior-Oriented

### **Configurations:**

486-Based PC's (Windows, Windows NT, Windows 95)

### **Description/Purpose:**

ProVision Workbench is a Business Object Modeling tool that combines Business Process (Re)engineering, Business Object Analysis and interfaces to client/server development tools in a single repository. ProVision Workbench will help define a business as a collection of interrelated objects, then implement these objects in a choice of popular client/server development environments. In a single repository, ProVision Workbench builds Business Interaction Models, Workflow Models, object-oriented Property and State Models - more then ten model types i all. Its features include a powerful on-line methodology, model interpreters and even a spell checker.



## ***QASE-RT by Advance System Technologies, Inc.***

### **Product Information:**

Version Number: 3.0  
Date of Last Release: March 1995  
Date of First Release: April 1990  
Number Sold: 30  
Single User Price: \$16,000 Mac  
\$33,700 UNIX

### **Contact Information:**

#### **Point of Contact:**

Name: Don Hazell  
Address: 12200 E. Briarwood Ave.  
Suite 260  
Englewood, CO 80112  
Phone : 303-790-4242 ext 129  
Fax : 303-790-2816  
E-mail : dhazell@advstech.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	✓ Client-Server	
✓ Other: Performance Analysis & Architecture Planning		

### **Intended Customers:**

All	Engineering	✓ Embedded Systems
✓ MIS	✓ Client-Server	✓ Real-time
Database	Other:	

### **Primary Methodology:**

Structured	Object-Oriented	Behavior-Oriented
------------	-----------------	-------------------

### **Configurations:**

Macintosh - Mac or PowerPC	Windows/NT- Intel PC
HP/UX - HP workstation	Solaris - SUN Sparc Workstation
AIX - RS6000	

### **Description/Purpose:**

QASE-RT is an analytic and simulation modeling tool for determining the capacity planning and response time requirements of distributed, client/server applications. AST's distributed computing group use QASE and end-to-end transaction process to build accurate models of the server, database, client, network, and application components of a distributed application. This can be used as powerful "what if" analysis regarding application /data placement, scalability and service level agreements

## ***Rational Rose by Rational Software Corp***

### **Product Information:**

Version Number: 2.5  
 Date of Last Release: November 1994  
 Date of First Release: January 1992  
 Number Sold: 10,000  
 Single User Price: PC - \$1,600  
 UNIX - \$4,800

### **Contact Information:**

#### **Point of Contact:**

Name: Loren Archer  
 Address: 2800 San Tomas Expressway  
 Santa Clara, CA 95051  
 Phone : 408-496-3600  
 Fax : 408-496-3974  
 E-mail : loren@rational.com

### **Lifecycle Phases and Activities:**

Requirements Specification	√ High Level Design	Re-engineering
√ Requirements Analysis	√ Low Level Design	√ Environment
Requirements Documentation	Database	√ System Analysis
√ Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

√ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

Structured      √ Object-Oriented      Behavior-Oriented

### **Configurations:**

PC (Windows), HP (HP/UX), IBM (AIX) and Sun (Sun OS and Solaris)

### **Description/Purpose:**

Rational Rose is a graphical software-engineering tool that supports object-oriented analysis and design, helping development teams produce object-oriented applications more effectively. With Rational Rose, you can represent, verify and communicate the analysis and design model of a software system.

## ***Rational Rose/Ada by Rational Software Corp***

### **Product Information:**

Version Number: 2.5  
Date of Last Release: January 1995  
Date of First Release: June 1994  
Number Sold: 1,000  
Single User Price: UNIX - \$8,400

### **Contact Information:**

#### **Point of Contact:**

Name: Loren Archer  
Address: 2800 San Tomas Expressway  
Santa Clara, CA 95051  
Phone : 408-496-3600  
Fax : 408-496-3974  
E-mail : loren@rational.com

### **Lifecycle Phases and Activities:**

Requirements Specification	√ High Level Design	√ Re-engineering
√ Requirements Analysis	√ Low Level Design	√ Environment
Requirements Documentation	Database	√ System Analysis
√ Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

√ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

Structured      √ Object-Oriented      Behavior-Oriented

### **Configurations:**

HP (HP/UX), IBM (AIX) and Sun (Sun OS and Solaris)

### **Description/Purpose:**

Rational Rose/Ada is a graphical software-engineering tool that supports object-oriented analysis, design and implementation. Helping development teams produce Ada applications more effectively. With Rational Rose/Ada, you can represent, verify and communicate the analysis and design of a software system. Additionally, you can take advantage of smart code generation and reverse-engineering capabilities for Ada.

## ***Rational Rose/C++ by Rational Software Corp***

### **Product Information:**

Version Number: 2.5  
 Date of Last Release: January 1995  
 Date of First Release: June 1994  
 Number Sold: 5,000  
 Single User Price: PC - \$2,400  
 UNIX - \$8,400

### **Contact Information:**

#### **Point of Contact:**

Name: Loren Archer  
 Address: 2800 San Tomas Expressway  
 Santa Clara, CA 95051  
 Phone : 408-496-3600  
 Fax : 408-496-3974  
 E-mail : loren@rational.com

### **Lifecycle Phases and Activities:**

✓ Requirements Specification	✓ High Level Design	✓ Re-engineering
✓ Requirements Analysis	✓ Low Level Design	✓ Environment
Requirements Documentation	Database	✓ System Analysis
✓ Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

Structured      ✓ Object-Oriented      Behavior-Oriented

### **Configurations:**

HP (HP/UX), IBM (AIX) and Sun (Sun OS and Solaris)

### **Description/Purpose:**

Rational Rose/Ada is a graphical software-engineering tool that supports object-oriented analysis, design and implementation. Helping development teams produce C++ applications more effectively. With Rational Rose/Ada, you can represent, verify and communicate the analysis and design model of a software system. Additionally, you can take advantage of smart code generation and reverse-engineering capabilities for C++.

## ***RDBMS by Cadre Technologies, Inc.***

### **Product Information:**

Version Number: 3.2  
Date of Last Release: December 1994  
Date of First Release: 1988  
Number Sold: 4,500+  
Single User Price: Discounts Available

### **Contact Information:**

#### **Point of Contact:**

Name: Doug Trolan, Dir Fed Ops  
Address: 6701 Democracy Plaza  
Suite 710  
Bethesda, MD 20817  
Phone : 301-897-4101  
Fax : 301-897-3106  
E-mail : dtrolan@cadre.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	✓ Re-engineering
✓ Requirements Analysis	Low Level Design	Environment
Requirements Documentation	✓ Database	System Analysis
Requirements Tracing/Mgmt.	✓ Client-Server	Other:

### **Intended Customers:**

All	Engineering	Embedded Systems
✓ MIS	Client-Server	Real-time
✓ Database	Other:	

### **Primary Methodology:**

✓ Structured	Object-Oriented	Behavior-Oriented
--------------	-----------------	-------------------

### **Configurations:**

Sparc(SunOS, Solaris), H-P PA RISC (HP-UX), IBM RS6000 (AIX), DEC ALPHA (OSF/1), DEC (OPEN VMS), Windows 3.1 (client planned), Windows 95 (client planned)

### **Description/Purpose:**

RDBMS Apps Dev is an integrated set of modeling tools for the development of information system applications for relational databases, specifically CA-Ingress, Infromix, Oracle or Sybase. The toolset includes support for four phases of applications development: Analyst, Architect, Designer and Programmer. Using the Yourdon methodology, the toolset supports the drawing and checking of entity relationship diagrams, data structure diagrams, data flow diagrams, and form sequence diagrams with a central repository. Entity relationship diagrams are transformed into SQL (including constraints and stored procedures) and structured design diagrams are transformed into C and embedded SQL or the 4GL of the target RDBMS. Prototyping is supported with the form sequence diagram.

## ***RoboChart by Digital Insight***

### **Product Information:**

Version Number: OL8.0 XM2.0  
 Date of Last Release: June 1995  
 Date of First Release:  
 Number Sold:  
 Single User Price: First Float License  
 \$850, each Additional \$395

### **Contact Information:**

#### **Point of Contact:**

Name: Lynn Morrison  
 Address: PO Box 533  
 Simi Valley, CA 93062-0533  
 Phone : 805-583-3627  
 Fax : 805-583-3809  
 E-mail : rc-sales@diginc.com

### **Lifecycle Phases and Activities:**

Requirements Specification	✓ High Level Design	✓ Re-engineering
Requirements Analysis	✓ Low Level Design	Environment
✓ Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured      ✓ Object-Oriented      Behavior-Oriented

### **Configurations:**

Sparc (SunOS/Solaris)

### **Description/Purpose:**

Interactive editor for flow diagrams (DFD, ERD, Flowchart, Org. Chart, etc). Supports hundreds of resizable shapes, dozens of hire types, fonts, colors. Supports hierarchical diagrams, custom commands, hypertext links, postscript, EPS, HPGL, MIS ASCII output formats. Rapid editing by direct manipulation; labels and flows are update automatically.

## ***RTM by Marconi Systems Technology***

### **Product Information:**

Version Number: 2.0  
Date of Last Release:  
Date of First Release:  
Number Sold:  
Single User Price:

### **Contact Information:**

#### **Point of Contact:**

Name: Doug Trolan, Dir Fed Ops  
Address: 6701 Democracy Plaza  
Suite 710  
Bethesda, MD 20817  
Phone : 301-897-4101  
Fax : 301-897-3106  
E-mail : dtrolan@cadre.com

### **Lifecycle Phases and Activities:**

√ Requirements Specification	High Level Design	Re-engineering
√ Requirements Analysis	Low Level Design	Environment
√ Requirements Documentation	Database	System Analysis
√ Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

√ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

√ Structured      √ Object-Oriented      √ Behavior-Oriented

### **Configurations:**

Sparc(SunOS, Solaris), H-P PA RISC (HP-UX), IBM RS6000 (AIX), DEC ALPHA (OSF/1), DEC (OPEN VMS), SGI(IRIX), Windows 3.1 (client planned)

### **Description/Purpose:**

Cadre Technologies, Inc. is a RTM distributor.  
A robust requirements management and traceability tool built on the Oracle database. RTM includes requirements stripper from source documents, robust requirements engineering capability, and easy to use report writer. RTM provides a customizable schema for requirements capture, tracing and impact analysis. Extensive support for system engineering is coupled with a bi-directional interface to Cadre's Teamwork and ObjectTeam families of CASE tools for software system engineering. Output formats supported include postscript, FrameMaker, Interleaf and RTF.

## SES/Objectbench by SES

### Product Information:

Version Number: 2.0  
 Date of Last Release: June 1995  
 Date of First Release: May 1993  
 Number Sold: 400  
 Single User Price: Editor = \$5000  
 Simulator = \$20,000  
 Code Generation/Arch. Development = Varies

### Contact Information:

#### Point of Contact:

Name: Marc Fackler  
 Address: 1870 Dublin, Suite 8  
 Colorado Springs, CO 80918  
 Phone : 719-522-9050  
 Fax : 719-522-9052  
 E-mail : fackler@ses.com

### Lifecycle Phases and Activities:

Requirements Specification	✓ High Level Design	Re-engineering
Requirements Analysis	✓ Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### Intended Customers:

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### Primary Methodology:

Structured	✓ Object-Oriented	Behavior-Oriented
------------	-------------------	-------------------

### Configurations:

Sun (Sun OS and Solaris), HP (HPUX), IBM RS/6000 (AIX)

### Description/Purpose:

SES/Objectbench is an object-oriented analysis and design modeling toolset. It includes a graphical editor that allows users to easily create and validate models. Models are dynamically checked for completeness and accuracy through user controlled simulation and on-screen animation. The SES/Objectbench OOD toolset fully supports architecture definition, development and code generation.



## ***SES/Workbench by SES***

### **Product Information:**

Version Number: 3.0  
Date of Last Release: December 1994  
Date of First Release: April 1989  
Number Sold: 1500  
Single User Price: \$48,000 for a single  
user floating license

### **Contact Information:**

#### **Point of Contact:**

Name: Marc Fackler  
Address: 1870 Dublin, Suite 8  
Colorado Springs, CO 80918  
Phone : 719-522-9050  
Fax : 719-522-9052  
E-mail : fackler@ses.com

### **Lifecycle Phases and Activities:**

Requirements Specification	√	High Level Design	√	Re-engineering
Requirements Analysis		Low Level Design		Environment
Requirements Documentation	√	Database	√	System Analysis
Requirements Tracing/Mgmt.	√	Client-Server		Other:

### **Intended Customers:**

√ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

Structured	Object-Oriented	Behavior-Oriented
------------	-----------------	-------------------

### **Configurations:**

Sun (Sun OS and Solaris), HP (HPUX), IBM RS/6000 (AIX)

### **Description/Purpose:**

SES/Workbench is an integrated modeling environment for capturing system design decisions and analyzing performance throughout a project's lifecycle. SES/Workbench supports graphical capture, animation and simulation of mission-critical systems, and includes comprehensive tracing, break, statistics, parameterization and library features for analyzing complex systems.

## **SoDA by Rational Software Corp**

### **Product Information:**

Version Number: 1.3  
 Date of Last Release: May 1995  
 Date of First Release: September 1994  
 Number Sold: 1,000  
 Single User Price: \$8,000

### **Contact Information:**

#### **Point of Contact:**

Name: Balaji Yelamanchili  
 Address: 2800 San Tomas Expressway  
 Santa Clara, CA 95051  
 Phone : 408-496-3600  
 Fax : 408-496-3974  
 E-mail : by@rational.com

### **Lifecycle Phases and Activities:**

Requirements Specification	✓ High Level Design	Re-engineering
Requirements Analysis	✓ Low Level Design	Environment
✓ Requirements Documentation	Database	System Analysis
✓ Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

✓ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured          ✓ Object-Oriented          ✓ Behavior-Oriented

### **Configurations:**

IBM (AIX) and Sun (Sun OS and Solaris)

### **Description/Purpose:**

SoDA automates the production of software documentation. SoDA provides the following major capabilities:

- Supports documentation automation throughout the lifecycle.
- Allows use of multiple sources of information to create a single document.
- Provide for checking consistency between documents, sources and information.
- Let's you customize document templates w/o programming.
- Provide support for requirements traceability.
- Provide document template for current DOD/Mil standards.

## ***SoftTest by Bender & Associates, Inc.***

### **Product Information:**

Version Number: 5.0  
Date of Last Release: May 1995  
Date of First Release: October 1987  
Number Sold: 300+  
Single User Price: \$2,500/copy

### **Contact Information:**

#### **Point of Contact:**

Name: Blaine Bragg  
Address: PO Box 849  
Larkspur, CA 94939  
Phone : 415-924-9196  
Fax : 415-924-3020  
E-mail : bbragg@softtest.com

### **Lifecycle Phases and Activities:**

√ Requirements Specification	High Level Design	Re-engineering
√ Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	√ Other: Testing

### **Intended Customers:**

√ All	Engineering	Embedded Systems
MIS	Client-Server	Real-time
Database	Other:	

### **Primary Methodology:**

√ Structured      √ Object-Oriented      √ Behavior-Oriented

### **Configurations:**

PC (DOS, Windows, OS/2) and UNIX (XWindows)

### **Description/Purpose:**

SoftTest is a functional test case design tool that generates the minimum yet sufficient set of tests required to ensure 100% functional coverage during testing.

Test cases designed using SoftTest SIGNIFICANTLY INCREASE THE EFFECTIVENESS of the testing process by increasing functional code coverage by as much as 60%! SoftTest also SIGNIFICANTLY INCREASES THE EFFICIENCY of testing by reducing the number of required functional test cases by up to one half!

SoftTest is the only tool available that utilizes a mathematically rigorous technique for designing and evaluating functional test cases. SoftTest aids in validating systems requirements by identifying most of the system's logical inconsistencies and all of the system's elementary functions. SoftTest then determines the necessary test cases that will ensure that 100% of the system's functionality is tested. SoftTest also aids in project management by providing functional coverage analysis and archiving of test libraries. SoftTest also provides quantitative metrics allowing manageable completion criteria for the test process.

## ***SUMMIT-D by Coopers & Lybrand L.L.P.***

### **Product Information:**

Version Number: 2.7  
 Date of Last Release: February 1995  
 Date of First Release: 1989  
 Number Sold: 80  
 Single User Price: \$60,000

### **Contact Information:**

#### **Point of Contact:**

Name: John J. Newcomb  
 Address: Princeton Forrestal Village  
 136-300 Main Street  
 Princeton, NJ 08540  
 Phone : 609-520-6131  
 Fax : 609-520-6195  
 E-mail :

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	✓ Environment
Requirements Documentation	Database	System Analysis
✓ Requirements Tracing/Mgmt.	Client-Server	
✓ Other: Process Management		

### **Intended Customers:**

All	Engineering	Embedded Systems
✓ MIS	✓ Client-Server	Real-time
✓ Database	Other:	

### **Primary Methodology:**

✓ Structured      ✓ Object-Oriented      Behavior-Oriented

### **Configurations:**

Windows and WIN/OS2

### **Description/Purpose:**

C&L's SUMMIT-D is a hypertext/hypergraphics-driven systems development methodology which covers the complete range of delivery scenarios in the development life cycle. SUMMIT-D is rich in content and highly accessible and usable.

## ***SUMMIT Process by Coopers & Lybrand L.L.P.***

### **Product Information:**

Version Number: 2.7  
Date of Last Release: February 1995  
Date of First Release: 1993  
Number Sold: 40  
Single User Price: \$50,000 (License for 50 copies)

### **Contact Information:**

#### **Point of Contact:**

Name: John J. Newcomb  
Address: Princeton Forrestal Village  
136-300 Main Street  
Princeton, NJ 08540  
Phone : 609-520-6131  
Fax : 609-520-6195  
E-mail :

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
Requirements Analysis	Low Level Design	√ Environment
Requirements Documentation	Database	System Analysis
√ Requirements Tracing/Mgmt.	Client-Server	
√ Other: Process Management		

### **Intended Customers:**

√ All	√ Engineering	Embedded Systems
√ MIS	√ Client-Server	Real-time
√ Database	Other:	

### **Primary Methodology:**

√ Structured      √ Object-Oriented      Behavior-Oriented

### **Configurations:**

Windows and WIN/OS2

### **Description/Purpose:**

C&L's SUMMIT Process software is a fully automated environment designed to clarify and improve the Information Technology development process. Working in concert with C&L's acclaimed SUMMIT-D systems development methodology, this innovative software helps to improve product and process quality, manage risk, meet user business needs and increase productivity. (components include workbenches for Planner, Developer and Admin.)

## ***System Engineer by LBMS, Inc.***

### **Product Information:**

Version Number: 6.0  
Date of Last Release: April 1995  
Date of First Release: May 1990  
Number Sold: 25,000  
Single User Price: \$5,000 per seat

### **Contact Information:**

#### **Point of Contact:**

Name: John Wills  
Address: 1800 West Loop South  
6th Floor  
Houston, TX 77027  
Phone : 800-231-7575  
Fax : 713-343-4419  
E-mail : sales@lbms.com

### **Lifecycle Phases and Activities:**

✓ Requirements Specification	✓ High Level Design	✓ Re-engineering
✓ Requirements Analysis	✓ Low Level Design	Environment
✓ Requirements Documentation	✓ Database	✓ System Analysis
✓ Requirements Tracing/Mgmt.	✓ Client-Server	Other:

### **Intended Customers:**

All	✓ Engineering	Embedded Systems
✓ MIS	✓ Client-Server	Real-time
✓ Database	Other:	

### **Primary Methodology:**

✓ Structured                      Object-Oriented                      ✓ Behavior-Oriented

### **Configurations:**

Novell Netware 4.x, Windows 3.1, Windows for Workgroups, Windows NT, and DOS

### **Description/Purpose:**

System Engineer is an industry leading client/server CASE tool. It provides a robust multi-user repository for concurrent team member modeling, an integrated set of requirements through design models, has an open and extensible repository, generates and reverse engineers relational databases, and can produce specifications documentation automatically using word processors and windows functionality like ODE and OLE. It also integrates seamlessly with development tools like PowerSoft, PowerSourcer and Microsoft's VisualBasic through its common repository.

## ***Teamwork by Cadre Technologies, Inc.***

### **Product Information:**

Version Number: 6.0  
Date of Last Release: September 1994  
Date of First Release: 1985  
Number Sold: 32,000  
Single User Price: GSA Available

### **Contact Information:**

#### **Point of Contact:**

Name: Doug Trolan, Dir Fed Ops  
Address: 6701 Democracy Plaza  
Suite 710  
Bethesda, MD 20817  
Phone : 301-897-4101  
Fax : 301-897-3106  
E-mail : dtrolan@cadre.com

### **Lifecycle Phases and Activities:**

- |                              |                     |                   |
|------------------------------|---------------------|-------------------|
| ✓ Requirements Specification | ✓ High Level Design | ✓ Re-engineering  |
| ✓ Requirements Analysis      | ✓ Low Level Design  | ✓ Environment     |
| ✓ Requirements Documentation | ✓ Database          | ✓ System Analysis |
| ✓ Requirements Tracing/Mgmt. | Client-Server       | Other:            |

### **Intended Customers:**

- |            |               |                    |
|------------|---------------|--------------------|
| All        | ✓ Engineering | ✓ Embedded Systems |
| MIS        | Client-Server | ✓ Real-time        |
| ✓ Database | Other:        |                    |

### **Primary Methodology:**

- |              |                 |                   |
|--------------|-----------------|-------------------|
| ✓ Structured | Object-Oriented | Behavior-Oriented |
|--------------|-----------------|-------------------|

### **Configurations:**

Sparc(SunOS, Solaris), H-P PA RISC (HP-UX), IBM RS6000 (AIX), DEC ALPHA (OSF/1), DEC (OPEN VMS), SGI(IRIX), DEC(ULTRIX), Intel (OS/2), Intel (Solaris).

### **Description/Purpose:**

The Teamwork Project Environment (TPE) provides a multi-user, software system engineering environment for requirements analysis and verification. The TPE provides the CASE repository that the Teamwork tools access and store information in. The TPE includes the data dictionary and various integration mechanisms such as ACCESS (the C or Ada based Application Programming Interface to the Teamwork Project Database), User Defined Menus, and Cadre CDIF (ASCII export and import mechanism). The Teamwork tools that access and store information in the TPE include: Teamwork/SA, Teamwork/RT, Teamwork/DA, Teamwork/SC, Teamwork/IMSQL, Teamwork/SD, the Teamwork/ADA suite of tools and the Ensemble suite of tools.

## ***Teamwork/ADA by Cadre Technologies, Inc.***

### **Product Information:**

Version Number: 6.0  
 Date of Last Release: September 1994  
 Date of First Release: 1991  
 Number Sold: 11,500+  
 Single User Price: GSA Available

### **Contact Information:**

#### **Point of Contact:**

Name: Doug Trolan, Dir Fed Ops  
 Address: 6701 Democracy Plaza  
 Suite 710  
 Bethesda, MD 20817  
 Phone : 301-897-4101  
 Fax : 301-897-3106  
 E-mail : dtrolan@cadre.com

### **Lifecycle Phases and Activities:**

Requirements Specification	✓ High Level Design	✓ Re-engineering
Requirements Analysis	✓ Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

All	✓ Engineering	✓ Embedded Systems
MIS	Client-Server	✓ Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured      ✓ Object-Oriented      ✓ Behavior-Oriented

### **Configurations:**

Sparc(SunOS, Solaris), H-P PA RISC (HP-UX), IBM RS6000 (AIX), DEC ALPHA (OSF/1), DEC (OPEN VMS), SGI(IRIX), DEC(ULTRIX)

### **Description/Purpose:**

Teamwork/Ada™ provides the capability to draw, edit and check Ada Structure Graphs (ASG's). The ASG's are then transformed to Ada code with the Ada Source Builder (ASB). The ASG's capture the packaging, tasking, and library level architecture of the Ada system design. Teamwork/ADA has been integrated with Ada-Assured from Grammatek, an Ada language sensitive editor. Teamwork/ADA is part of RevAda/Twk, Cadre's Ada reverse engineering tool. Cadre also offer QualGen from SSD, a tool for capturing metrics about Ada code.



## ***Teamwork/Dynamic Analysis by Cadre Technologies, Inc.***

### **Product Information:**

Version Number: 6.0  
Date of Last Release: September 1994  
Date of First Release: 1991  
Number Sold: 50  
Single User Price: GSA Available

### **Contact Information:**

#### **Point of Contact:**

Name: Doug Trolan, Dir Fed Ops  
Address: 6701 Democracy Plaza  
Suite 710  
Bethesda, MD 20817  
Phone : 301-897-4101  
Fax : 301-897-3106  
E-mail : dtrolan@cadre.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
√ Requirements Analysis	Low Level Design	Environment
Requirements Documentation	Database	√ System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

All	√ Engineering	√ Embedded Systems
MIS	Client-Server	√ Real-time
Database	Other:	

### **Primary Methodology:**

√ Structured                      Object-Oriented                      √ Behavior-Oriented

### **Configurations:**

Sparc(SunOS, Solaris), H-P PA RISC (HP-UX), IBM RS6000 (AIX), DEC ALPHA (OSF/1), DEC (OPEN VMS), SGI(IRIX), DEC(ULTRIX), Intel (Solaris).

### **Description/Purpose:**

In conjunction with Teamwork/SA and Teamwork/RT, Teamwork/Dynamic Analysis provides the capability to model system performance. Teamwork/DA verifies the dynamic behavior of real-time system specifications using token-based symbolic simulation of the processes and state information on the DFD's and STD's created using Teamwork/SA and Teamwork/RT. Teamwork/DA is useful for discovering resource contention, race and deadlock conditions for complex mission critical systems. Teamwork/DA puts in motion the static diagrams by passing tokens between processes and marking system states. The interactive mode provides breakpoints, trace, and performance statistics. Priority and pre-emption schemes can be simulated.

## ***Teamwork/IMSQL by Cadre Technologies, Inc.***

### **Product Information:**

Version Number: 6.0  
 Date of Last Release: September 1994  
 Date of First Release: 1986  
 Number Sold: 11,000  
 Single User Price: GSA Available

### **Contact Information:**

#### **Point of Contact:**

Name: Doug Trolan, Dir Fed Ops  
 Address: 6701 Democracy Plaza  
 Suite 710  
 Bethesda, MD 20817  
 Phone : 301-897-4101  
 Fax : 301-897-3106  
 E-mail : dtrolan@cadre.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	Re-engineering
✓ Requirements Analysis	Low Level Design	Environment
Requirements Documentation	✓ Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

All	✓ Engineering	Embedded Systems
✓ MIS	Client-Server	Real-time
✓ Database	Other:	

### **Primary Methodology:**

✓ Structured	Object-Oriented	Behavior-Oriented
--------------	-----------------	-------------------

### **Configurations:**

Sparc(SunOS, Solaris), H-P PA RISC (HP-UX), IBM RS6000 (AIX), DEC ALPHA (OSF/1), DEC (OPEN VMS)[Teamwork/IM only], SGI(IRIX)[Teamwork/IM only], DEC(ULTRIX), Intel (OS/2)[Teamwork/IM only], Intel (Solaris).

### **Description/Purpose:**

Teamwork/IMSQL™ provides the capability to draw, edit, and check entity relationship diagrams (ERD's). Support Chen notation. Comes integrated with the Teamwork Project Environment (TPE) data dictionary. Checking includes syntax, balancing and consistency. The SQL portion of the product transforms the ERD's to ANSI standard SQL (DDL) to assist in generating the RDBMS schema.

STSC RED Product Sheet Version 2.0c

## ***Teamwork/RT by Cadre Technologies, Inc.***

### **Product Information:**

Version Number: 6.0  
Date of Last Release: September 1994  
Date of First Release: 1985  
Number Sold: 32,000  
Single User Price: GSA Available

### **Contact Information:**

#### **Point of Contact:**

Name: Doug Trolan, Dir Fed Ops  
Address: 6701 Democracy Plaza  
Suite 710  
Bethesda, MD 20817  
Phone : 301-897-4101  
Fax : 301-897-3106  
E-mail : dtrolan@cadre.com

### **Lifecycle Phases and Activities:**

✓ Requirements Specification	✓ High Level Design	Re-engineering
✓ Requirements Analysis	Low Level Design	Environment
✓ Requirements Documentation	Database	✓ System Analysis
✓ Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

All	✓ Engineering	✓ Embedded Systems
MIS	Client-Server	✓ Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured                      Object-Oriented                      ✓ Behavior-Oriented

### **Configurations:**

Sparc(SunOS, Solaris), H-P PA RISC (HP-UX), IBM RS6000 (AIX), DEC ALPHA (OSF/1), DEC (OPEN VMS), SGI(IRIX), DEC(ULTRIX), Intel (OS/2), Intel (Solaris).

### **Description/Purpose:**

Teamwork/RT™ provides the capability to draw, edit, and check control flow diagrams (CFD's) and control specifications, namely State Transition Diagrams (STD's), Decision Tables (DT's), State Event Matrices (SEM's) and Process Activation Tables (PAT's). Supports Haatley/Pirbhai structured system specifications. Comes integrated with the Teamwork Project Environment (TPE) data dictionary. Checking includes syntax, balancing and consistency. The BANG metric is provided to measure the change in complexity of subsequent CFD's.

## ***Teamwork/SA by Cadre Technologies, Inc.***

### **Product Information:**

Version Number: 6.0  
 Date of Last Release: September 1994  
 Date of First Release: 1985  
 Number Sold: 32,000  
 Single User Price: GSA Available

### **Contact Information:**

#### **Point of Contact:**

Name: Doug Trolan, Dir Fed Ops  
 Address: 6701 Democracy Plaza  
 Suite 710  
 Bethesda, MD 20817  
 Phone : 301-897-4101  
 Fax : 301-897-3106  
 E-mail : dtrolan@cadre.com

### **Lifecycle Phases and Activities:**

√ Requirements Specification	√ High Level Design	Re-engineering
√ Requirements Analysis	Low Level Design	Environment
√ Requirements Documentation	Database	√ System Analysis
√ Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

All	√ Engineering	√ Embedded Systems
MIS	Client-Server	√ Real-time
Database	Other:	

### **Primary Methodology:**

√ Structured	Object-Oriented	Behavior-Oriented
--------------	-----------------	-------------------

### **Configurations:**

Sparc(SunOS, Solaris), H-P PA RISC (HP-UX), IBM RS6000 (AIX), DEC ALPHA (OSF/1), DEC (OPEN VMS), SGI(IRIX), DEC(ULTRIX), Intel (OS/2), Intel (Solaris).

### **Description/Purpose:**

Teamwork/SA™ provides the capability to draw, edit, and check data flow diagrams (DFD's) and decompose processes to process specifications (p-specs). Supports DeMarco functional decomposition for system specifications. Comes integrated with Teamwork Project Environment (TPE) data dictionary. Checking includes syntax, balancing and consistency. The BANG metric is provided to measure the change in complexity of subsequent DFD's.

## ***Teamwork/SD by Cadre Technologies, Inc.***

### **Product Information:**

Version Number: 6.0  
Date of Last Release: September 1994  
Date of First Release: 1987  
Number Sold: 20,000  
Single User Price: GSA Available

### **Contact Information:**

#### **Point of Contact:**

Name: Doug Trolan, Dir Fed Ops  
Address: 6701 Democracy Plaza  
Suite 710  
Bethesda, MD 20817  
Phone : 301-897-4101  
Fax : 301-897-3106  
E-mail : dtrolan@cadre.com

### **Lifecycle Phases and Activities:**

Requirements Specification	High Level Design	√ Re-engineering
Requirements Analysis	√ Low Level Design	Environment
Requirements Documentation	Database	System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

All	√ Engineering	√ Embedded Systems
MIS	Client-Server	√ Real-time
Database	Other:	

### **Primary Methodology:**

√ Structured                      Object-Oriented                      Behavior-Oriented

### **Configurations:**

Sparc(SunOS, Solaris), H-P PA RISC (HP-UX), IBM RS6000 (AIX), DEC ALPHA (OSF/1), DEC (OPEN VMS), SGI(IRIX), DEC(ULTRIX), Intel (OS/2), Intel (Solaris).

### **Description/Purpose:**

Teamwork/SD™ provides the capability to draw, edit, and check structure charts (SC's) and expand them to module specification (m-specs). Support Yourdon/Constantine structured design. Comes integrated with the Teamwork Project Environment (TPE) data dictionary. Checking includes syntax, balancing and consistency. Design metrics can be calculated based on the Structure Charts. Teamwork/SD is integrated with Cadre's Ensemble suite of C reverse engineering tools.

## ***TurboCASE by StructSoft, Inc.***

### **Product Information:**

Version Number: V4.25  
 Date of Last Release: April 1995  
 Date of First Release: September 1989  
 Number Sold: ~5000  
 Single User Price: \$995.00

### **Contact Information:**

#### **Point of Contact:**

Name: Shang Chyou  
 Address: 5416 56th Ave. SE  
 Bellevue, WA 98006  
 Phone : 206-644-9834  
 Fax : 206-644-7714  
 E-mail : 76636.254@compuserve.com

### **Lifecycle Phases and Activities:**

✓ Requirements Specification	✓ High Level Design	Re-engineering
✓ Requirements Analysis	✓ Low Level Design	Environment
✓ Requirements Documentation	✓ Database	✓ System Analysis
Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

All	✓ Engineering	✓ Embedded Systems
MIS	Client-Server	✓ Real-time
✓ Database	Other:	

### **Primary Methodology:**

✓ Structured      ✓ Object-Oriented      ✓ Behavior-Oriented

### **Configurations:**

All Macintosh, Systems 6 or 7

### **Description/Purpose:**

TurboCASE is an integrated, multi-window front-end CASE tool which supports methodologies ranging from the structured to the newer object-oriented software development. It can be used to model or develop client/server applications using one of the methodologies it supports. TurboCASE supports structured analysis with real-time extensions, data modeling with object-oriented analysis, structured design and object-oriented design. By using a repository, TurboCASE integrates all diagrams, allowing multiple views into the model you are creating. As you change one aspect of the model, other aspects are automatically changed. TurboCASE is more than just a documentation package. It provides the information needed to make intelligent modeling choices, and features built-in consistency checking and support for top-down or bottom-up modeling. With the newly added object-oriented supports, it lets you migrate to the newer software development methodologies.

## ***TurboCASE/Sys by StructSoft, Inc.***

### **Product Information:**

Version Number: V2.0  
Date of Last Release: January 1995  
Date of First Release: October 1992  
Number Sold: ~1000  
Single User Price: \$3,495.00

### **Contact Information:**

#### **Point of Contact:**

Name: Shang Chyou  
Address: 5416 56th Ave. SE  
Bellevue, WA 98006  
Phone : 206-644-9834  
Fax : 206-644-7714  
E-mail : 76636.254@compuserve.com

### **Lifecycle Phases and Activities:**

✓ Requirements Specification	High Level Design	Re-engineering
✓ Requirements Analysis	Low Level Design	Environment
✓ Requirements Documentation	Database	✓ System Analysis
✓ Requirements Tracing/Mgmt.	Client-Server	Other:

### **Intended Customers:**

All	✓ Engineering	✓ Embedded Systems
MIS	Client-Server	✓ Real-time
Database	Other:	

### **Primary Methodology:**

✓ Structured      Object-Oriented      ✓ Behavior-Oriented

### **Configurations:**

All Macintosh, Systems 6 or 7

### **Description/Purpose:**

TurboCASE/Sys supports the Hatley/Pirbhai system engineering methodology described in the book "Strategies for Real Time System Specification" published by Dorset House. It is the only tool of its kind that supports the whole methodology. It helps the system engineers to create and maintain not only the system requirements but also the architecture modules of the system. By enhancing the requirements model with technology dependent processing, the engineers can subdivide the systems into architecture modules. Each of these architecture models will have well-defined interfaces defined in the architecture dictionary. Using the tool iteratively on each architecture module, manageable system components can be identified, designed and implemented. Its integrated views of the system models and the automation of documentation generation simplifies the system engineering task greatly.

**Appendix C: Upper CASE Products – Product  
Critique Format**



This appendix contains the format of the Upper CASE product critique. User critiques are written by actual product users. Editing by the STSC will be kept to an absolute minimum. A critique has five sections: (1) Name, (2) Evaluation Context, (3) Strengths and Weaknesses, (4) Additional Comments, and (5) Vendor Comments. The purpose of the evaluation context section is to give a critique reader an idea of the biases of the evaluator and the applicability of the critique to the reader's context. The strengths and weaknesses, and additional comments sections allow the evaluator free-form commentary about the product. The vendor comments section allows a response by the product vendor.

The STSC actively requests and uses product critiques from experienced product users. These critiques highlight the experiences (good and bad) of actual product users. If you are a user of a product that is or should be in the product list and would like to write a critique, please contact the STSC. An example product critique is found in Appendix C, Upper CASE Product Critique Format. Please contact the Upper CASE team for critiques.

## Tool Critique

### Reviewer's Profile:

Company Profile: \_\_\_\_\_

Position/Title: \_\_\_\_\_

Main Duties: \_\_\_\_\_

\_\_\_\_\_

Years of software experience: \_\_\_\_\_

Years of experience with the tool: \_\_\_\_\_

Last time tool used:

Currently      6 months   1 year   >1 year

I am a software:

Manager      Engineer   Programmer      Novice

Date of Review: \_\_\_\_\_

### Tool Profile:

Vendor: \_\_\_\_\_

Version: \_\_\_\_\_

Hardware platform: \_\_\_\_\_

Operating system: \_\_\_\_\_

Memory used: \_\_\_\_\_

Disk space used: \_\_\_\_\_

Enhancements: \_\_\_\_\_

(accelerator, large monitor, graphics card, etc.)

Overall impression of the tool?

Excellent      Good      Fair      Poor

Quality of vendor support?

Excellent      Good      Fair      Poor      Unknown

### Project Information

### Notable Strength(s) of the Tool

### Notable Weakness(es) of the Tool

### Advice for Potential Buyers of this Tool

### Vendor Response:

(This page is intentionally left blank.)

## **Appendix D: Upper CASE Product Characteristics**

## D.1 Functional Upper CASE Tool Characteristics

The following sections identify the functional capabilities of Upper CASE tools. The evaluation of the functional capability of Upper CASE tools assesses the tool's capabilities. The STSC has categorized the functional capabilities of Upper CASE tools into eight main areas. These areas are delineated in Table D-1, Upper CASE Functional Areas. Each of these areas has even more detailed characteristics. For instance, the Model Analysis area includes consistency analysis, completeness analysis, behavior analysis, and four other tool characteristics. The complete list of these detailed characteristics are the level of abstraction at which tool selectors need cognizance, because it is at this level of detail that the selector specifies functionality requirements.

Information Capture	Data Repository
Methodology Support	Documentation
Model Analysis	Data Import/Export and Standardization
Requirements Tracing	Reusability Support

Table D-1. Upper CASE Tool Products

### D.1.1 Information Capture

The information capture functionality area deals with what types of information the tool is capable of handling. This information can be captured in a number of ways. The important idea is what type of information is captured, not how it is captured. Table D-2, Upper CASE Tool Information Types, lists the types of information that Upper CASE tools capture.

- System Function Descriptions
- Data Descriptions of System Functions Interfaces
- Data Descriptions of System Input/Output Device Interfaces
- System Logical Behavior
- System Timing Behavior
- Hardware/Software Context
- Software Architectural Structure
- Software Process Definitions
- Software Data Structures
- Software Process Control
- Software Process Concurrency
- Software Interprocess Data Communication
- Software Interprocess Synchronization

Table D-2. Upper CASE Tool Information Types

### D.1.2 Methodology Support

Methodology is the process the tool user follows to systematically develop correct and complete work products. A number of methodologies exist for requirements analysis and software design. The important ones that have been automated are listed in Table D-3, Upper CASE Methodologies.

- |                                    |                           |
|------------------------------------|---------------------------|
| • Real-Time Structured Development | • PAMELA                  |
| • Structured Analysis              | • ESML                    |
| • Structured Design                | • ADARTS                  |
| • Hatley/Pirbhai Extensions        | • PAISLey                 |
| • Object-Oriented Design           | • VDM                     |
| • Ada-Based Object-Oriented Design | • Petri Nets              |
| • Object-Oriented Analysis         | • Statecharts             |
| • Entity Relationship Modeling     | • Axiomatic Specification |

Table D-3. Upper CASE Methodologies

These methodologies require that various work products be created by the user. Since different methodologies can require the same work products, the products are listed separately in Table D-4, Upper CASE Tool Products.

- |                                |                                     |
|--------------------------------|-------------------------------------|
| • Data Flow Diagrams           | • Ada Package Dependency Diagrams   |
| • System Context Diagrams      | • Structure Charts                  |
| • Block Diagrams               | • Flowcharts                        |
| • Control Flow Diagrams        | • Screen and Report Diagrams        |
| • Entity Relationship Diagrams | • User-Tailored Diagrams            |
| • State Transition Tables      | • Object Diagrams                   |
| • Petri Net Diagrams           | • Object Hierarchy or Tree Diagrams |
| • Architecture Diagrams        | • Object Interaction Diagrams       |

Table D-4. Upper CASE Tool Products

### D.1.3 Model Analysis

The model analysis functionality area captures the techniques the tool uses to analyze the inputs. These techniques can be static or dynamic. They are used to prove qualities about the input requirements or specifications such as completeness or consistency. They are also

used to simulate the inputs at an early stage in the lifecycle. The important techniques are listed in Table D-5, Upper CASE Analysis Techniques.

- |   |   |
|---|---|
| <ul style="list-style-type: none"><li>• Consistency Checking</li><li>• Completeness Checking</li><li>• Data Normalization Analysis</li><li>• Man/machine Interface Analysis</li></ul> | <ul style="list-style-type: none"><li>• Behavior Analysis</li><li>• Scenario-Based Analysis</li><li>• Exhaustive Model Analysis</li></ul> |
|---|---|

Table D-5. Upper CASE Analysis Techniques

#### D.1.4 Requirements Tracing

The requirements tracing functionality area captures the attributes associated with the tracing of requirements between software lifecycle phases. Requirements tracing is important because it facilitates the management of interlifecycle dependencies. The important attributes are listed in Table D-6, Requirements Tracing Attributes.

- |   |
|---|
| <ul style="list-style-type: none"><li>• Extraction of Requirements from System and Software Documentation</li><li>• Inputs From Electronically Scanned Hard-Copy</li><li>• Multiple Requirements Baselines</li><li>• Tracing of System Requirements to Software Requirements</li><li>• Tracing of System Design Specifications to Software Requirements</li><li>• Tracing of Requirements to Software Design</li><li>• Tracing of Requirements to Source Code</li><li>• Tracing of Requirements to Software and System Test</li></ul> |
|---|

Table D-6. Requirements Tracing Attributes

#### D.1.5 Data Repository

The data repository functionality area captures the attributes associated with the database the tool uses. Most tools use proprietary databases. The database model presented to the user, the user interface to the database, and the extent to which the database can represent



software objects is critical to the database's overall functionality. The important attributes are listed in Table D-7, Upper CASE Data Repository Functional Attributes.

• Data Repository	• Contain Project Information
• Relational Database Type	• Contain Requirements Documents
• Object Database Type	• Contain Design Specifications
• Support both Text and Graphics	• Contain Source Code
• Query Capability	• Contain Test Descriptions and Procedures
• Access Control Capability	• Capacity Artificially Limited
• Concurrent Access to Entities	• Support Interactive Cross-Referencing
Configuration Management Capability	

Table D-7. Upper CASE Data Repository Functional Attributes

#### D.1.6 Documentation

The documentation functionality area captures the attributes associated with the documentation the tool produces. The important attributes are listed in Table D-8, Upper CASE Documentation Functional Attributes.

• Support Graphics/Text Integration	• Automatic Generation of Documentation
• Completely Compile a Document	• Rapid Draft Hard Copy
• On-Line Templates	• Interface to Other Document Generators
• 2167A Documentation Standard	• Desktop Publishing Interface

Table D-8. Upper CASE Documentation Functional Attributes

#### D.1.7 Data Import/Export and Standardization

The data import/export functionality area captures the attributes associated with how easily the tool can exchange data with other tools including other tools in the tool vendor's tool set. The important attributes are listed in Table D-9, Upper CASE Data Input/Output Functional Attributes.

- |   |
|---|
| <ul style="list-style-type: none"><li>• Between Toolkit Components (Intraoperability)</li><li>• With Other Tools (Interoperability)</li><li>• CAIS-A Interface Standards/Protocols Supported</li><li>• PCTE Compliance</li><li>• Compatibility with the evolving I-CASE environment</li></ul> |
|---|

Table D-9 Upper CASE Data I/O Functional Attributes

The Common Ada Programming Support Environment (APSE) Interface Set (CAIS) is an environment-defining standard that has been popular with the military. Recently, the U.S. Departments of Defense and Commerce have shown interest in another CASE tool integration framework. They have shown support for the establishment of a North American Forum to supplement the Portable Common Tools Environment (PCTE) standards effort of the European Computer Manufacturers Association (ECMA). In addition, the National Institute of Standards and Technology (NIST) has sponsored an initiative to support a proposed CASE tool framework. NIST recommends that all government agencies support the PCTE framework, although it has not yet been adopted as standard.

The NIST/ECMA reference model supports both horizontal and vertical integration. Horizontal (methodological) integration ensures the consistency of information within each life-cycle phase when many modeling methods are used (such as data, process, event-driven, and object-oriented). Vertical (full lifecycle) integration ensures the consistency of information generated in the many lifecycle phases.

The Department of Defense (DoD) is supporting the development of a standard and integrated engineering environment with the I-CASE (Integrated Computer-Aided Software Engineering) program. I-CASE will provide engineers with a standard environment that will remain the same across services and across DoD agencies. I-CASE includes tools for business case analysis, requirements analysis, prototyping, and code generation. A central repository will be used to store tool data so that as technology is upgraded, an engineer can simply remove the older tool and put the new one in place. The draft request for I-CASE proposals was released by the Air Force in March 1992.

#### **D.1.8 Reusability Support**

The reusability functionality area captures the attributes associated with how the tool supports reuse. The one attribute in this area deals with support for library design components.

### **D.2 Quality**

The following sections define and discuss the Upper CASE implications of the 12 quality attributes identified in the analysis phase. The quality attributes have dual implications, one for the Upper CASE tool and the other for its product. The Upper CASE tool and the product are considered distinct. A compiler is similar in that not only should the compiler itself be considered but the output (product) must also be considered. A fast compiler that produces slow executables has little value.

#### **D.2.1 Efficiency**

Upper CASE tool efficiency refers to the amount of utilization of a resource on a problem, using the Upper CASE tool. The three resources that need to be assessed are: (1) processor (time to complete a task), (2) memory (the secondary storage requirements to complete a task), and (3) communication (Input/Output and network considerations for multiprocessor systems or multiuser problems). For Upper CASE tools, efficiency is not expressed absolutely. Instead, it is expressed qualitatively in terms of acceptable, barely acceptable, or unacceptable. Several problems covering a range of sizes from small to large across each of the resources need to be assessed. When the tool performs adequately for a specific problem with respect to a particular resource, its efficiency is acceptable for that problem size and that resource. Barely acceptable performance occurs when the performance is acceptable but there is no room for performance growth.

Efficiency, as it applies to the products of Upper CASE tools, is not important because the products of these tools are paper reports. That the tools may support efficiency studies of their products, e.g., timing analysis of designs, is a matter of functionality and not quality.

### **D.2.2 Integrity**

Integrity concerns either software security failures due to unauthorized access or the corruption of the database. As a policy, the tool users should lose confidence in the integrity of the database if unauthorized access is allowed. Database corruption may be caused by such actions as legal but partial or inconsistent operations and erroneous but allowed operations.

The integrity of the products of the tool is a non-issue. Accessibility to the products is usually governed by the operating system of the developmental machine and never by the tool. Once a product has been produced, it is no longer part of the database and can no longer be corrupted.

### **D.2.3 Reliability**

Reliability concerns software failures. Reliability is normally measured by direct testing and analysis of error reports. With commercial software, direct testing is not feasible, and detailed error reports are not normally published. For Upper CASE tools, instead of directly measuring reliability, indicators such as maturity, published error reports, size of executable code, and errors uncovered during testing will be used.

Since the products of the Upper CASE tools are intermediate products of the entire software development process, their reliability cannot be tested.

### **D.2.4 Survivability**

Survivability deals with the ability of the software to perform even when portions of the system have failed. This issue is usually not important in the evaluation of Upper CASE tools because the greater issue of system availability is not critical in an office environment. However, if the tool uses different hardware resources, i.e., networked workstations with a file server, the issue of how the tool handles hardware resource failure, i.e., file server shutdown must be addressed.

Survivability is not an issue for the tool products because they are reports.

#### **D.2.5 Usability**

Usability is the extent to which resources required to acquire, install, learn, operate, prepare input for, and interpret output of the tool or the tool products are minimized. This attribute is probably the most important and critical quality attribute for which Upper CASE tools are evaluated. This is the quality attribute in which tool vendors differentiate themselves through such quality criteria as user interfaces, user documentation, and training.

The usability of the products of the tool is not an important quality issue. The ability to customize reports is addressed in the documentation portion of the functional capabilities of the tool.

#### **D.2.6 Correctness**

Correctness is the extent to which software design and implementation conform to specifications and standards. The correctness of a tool is evaluated in other portions of the evaluation framework, namely the functional capability area. The reliability quality attribute addresses known errors.

The correctness of the tool products is important. The generated products should conform to the specification captured by the tool.

#### **D.2.7 Maintainability**

Maintainability is the ease of effort to locate and fix software failures within a specified time period. This attribute is not of importance to the tool user; instead, the time and ability for the vendor to deliver software maintenance is important. The tool user is not concerned with the effort required to perform these actions. This time is addressed in the vendor information portion of the evaluation framework (under management concerns).

This attribute is of importance to the the tool's user of the products, but not to the tool user. The tool products should possess the quality attributes of maintainability.

#### **D.2.8 Verifiability (a.k.a. testability)**

Verifiability deals with the design characteristics that facilitate the testing of the tool or the tool's products. The testing of the tool is important to the developers of the tool but not to the tool user (except that a well-tested tool will have higher reliability).

The ability to test the tool's products is important in determining the quality of the tool. But for Upper CASE tools, testing is best addressed as a functional capability of the tool.

#### **D.2.9 Expandability (a.k.a. flexibility)**

Expandability is the ease in which current functions can be enhanced or new functions added. Flexibility is defined as the ease with which the software can be changed to meet other new requirements. Within the scope of evaluating Upper CASE tools and Upper CASE tool products, where the viewpoint is user-implemented changes (not developer-implemented changes), these attributes are dealt with in the reusability quality attribute.

#### **D.2.10 Interoperability**

Interoperability is the ability of separate systems to exchange database objects and their relationships without conversion. This is an important area, capturing if, how much, and how well the Upper CASE tool implements data exchange standards. This area is addressed in the Functional Capabilities portion of the evaluation framework. It is not an important quality attribute for either Upper CASE tools or their products.

#### **D.2.11 Reusability**

Reusability is the extent to which a component can be adapted for use in another application. Within the scope of evaluating Upper CASE tools, reusability deals with how easily the tool can be used for other projects.

The issue of reusability of the products of the CASE tool is dealt with in the functional capabilities portion of the evaluation framework.

#### **D.2.12 Transportability (a.k.a. Portability)**

Transportability is the ability of a software item to be installed in a different environment without change in functionality. Within the scope of evaluating Upper CASE

tools, it deals with how many platforms and operating systems with which the tool works. This area is addressed in the portion of the evaluation framework that deals with operational constraints.

This is a non-issue with Upper CASE tool products since, by their very nature, they are reports not associated with any particular environment.

## **Appendix E: Recommended Readings**



## 1 General Selection Process

The papers in this section describe the process others have used to select and evaluate CASE tools. They emphasize the selection process and not selection criteria.

- 1-1 IEEE, p1209 Standard, "Recommended Practice for the Evaluation and Selection of CASE Tools."

This standard provides a recommended practice to evaluate and select CASE tools. This standard was primarily developed to provide guidance to evaluate and select tools used to support software engineering activities as opposed to general purpose tools, e.g., word processors, spread sheets, that may incidentally be used in support of software engineering activities.

## 2 General Management

These papers and reports contain information that is useful to managers needing an overview of the benefits and issues associated with Upper CASE software.

- 2-1 Bouldin, B., "Agents of Change: Managing the Introduction of Automated Tools," Prentice Hall, 1989.

This book contains a lot of good advice that is based upon the implementation of the CASE tool Excelerator at Bell Labs.

- 2-2 Brooks, F., "The Mythical Man-Month," Addison Wesley Publishing Co., 1978.

This best seller should be read by every project manager. At IBM, Brooks was the manager of the OS 360 project.

- 2-3 Gilb, T., "Principles of Software Engineering Management," Addison Wesley Publishing Company, 1988.

This book contains interesting ideas on quantifying qualitative goals. It advocates an extreme strategy for incremental delivery that makes a lot of sense.

- 2-4      Humphrey, W.S., "CASE Planning and the Software Process," CMU/SEI-89-TR-26, Software Engineering Institute, 1989.  
This paper addresses the potential benefits and costs of CASE. The benefits and costs are broken down by an organization's level in the SEI Software Process Maturity Model. Brief management overviews and remedies of potential problems in the acquisition, installation, and use of CASE are given.
- 2-5      Humphrey, W.S., *"Managing the Software Process,"* Addison-Wesley, 1989.  
Textbook that deals with understanding and managing the software process of an organization. This text summarizes the process maturity work at the SEI and provides practical guidance for assessing and improving the software development and maintenance process.
- 2-6      Yourdon, E., "Managing the System Life Cycle," Prentice Hall, 1988 (2nd edition).  
A short, easy to read, minor rewrite of a much older book.

### 3      Evaluation/Selection Criteria

The papers in this section contain listings of evaluation and selection criteria that can be used.

- 3-1      Bowen, T.P., et al., "Specification of Software Quality Attributes: Software Quality Evaluation Guidebook," RADC-TR-85-37, Vol. III, Rome Air Development Center, February 1985.  
This report contains definitions of software quality and how to measure it. The measurements are performed during software creation (different measurements at different parts of the lifecycle) with the goal of engineering software quality into the product. The Evaluation and Validation (E&V) manuals have an updated list of quality factors, criteria, and metrics. This is the original source. Its discussion of the attribute models is much more complete than the E&V discussion.

### 4      Methodologies

These papers and books describe widely used and popular Upper CASE methodologies.

- 4-1 Booch, G., "*Object Oriented Design With Applications*," Prentice Hall, 1991.  
This excellent book discusses issues that relate to every major object-oriented language. A new notation is introduced for OOD.
- 4-2 Buhr, R., "*Practical Visual Techniques in System Design*," Englewood Cliffs, NJ, Prentice-Hall, 1990.  
This book describes a method to design behavior-intensive systems (embedded, real-time, multitasking, multiprocessing, distributed) based on machine charts. Machine charts are a notational technique that supports the modeling of system behavior.
- 4-3 Chen, P., "*Entity-Relationship Approach to Systems Analysis and Design*", North Holland Publishing Company, 1980.  
This book covers entity relationship diagrams as well as entity relationship modeling.
- 4-4 Coad, P. and E. Yourdon, "*Object-Oriented Analysis*," Englewood, NJ, Yourdon Press Computing Series, Prentice-Hall, 1990.  
This book introduces object-oriented analysis, an analysis technique-based objects and attributes, classes and members, and wholes and parts. This is not a standardized method; rather, the book provides a starting point for using OOA within your particular organization, tailoring it to meet your specific needs.
- 4-5 Constantine, L. and E., Yourdon, "*Structured Design*," Prentice Hall, 1975.  
This book gives the most thorough treatment of the subject.
- 4-6 DeMarco, T., "*Structured Analysis and System" Specification*," Englewood Cliffs, N. J., Prentice-Hall, 1978.  
This book covers structured analysis or the specification of systems using a methodology that is an extension of the Structured Design methodology that was proposed by Edward Yourdon and Larry Constantine.
- 4-7 Fleming, C. and B. von Halle, "*Handbook of Relational Database Design*," Addison Wesley, 1989.  
Procedures are given in a step-by-step fashion for producing a logical data model (ERD) and also for translating that model into a set of tuned relational database structures. The book is very comprehensive.

- 4-8     Hatley, D. and I. Pirbhai, "*Strategies for Real-Time System Specification*," New York City, N.Y., Dorset House Publishing Co., 1988.  
       This book covers a methodology to model the requirements and architecture of real-time systems. It integrates a finite-state machine structure into the classical structured analysis methods.
- 4-9     Hawryszkiewicz, I., "*Database Analysis and Design*," MacMillan Publishing Co., 1984.  
       Relational theory and entity relationship diagrams are treated in this good textbook. DBMS and implementation issues are also discussed.
- 4-10    Jackson, M., "*Principles of Program Design*," Academic Press, 1975.  
       This book presents an approach to software design that is data structure oriented.
- 4-11    Martin, J., "*Information Engineering, Book 3 - Design and Construction*," Prentice Hall, 1989.  
       This book deals with James Martin's Information Engineering Methodology.
- 4-12    Martin, J., "*Strategic Information Planning Methodologies*," Prentice Hall, 1989 (2nd edition).  
       The first edition (1982) has undergone a minor revision. Business Systems Planning is treated via an excellent discussion.
- 4-13    Mellor, S. and S. Shlaer, "*Object Lifecycles: Modeling the World in States*," Prentice Hall, 1992.  
       Introduction to Shlaer/Mellor object oriented analysis.
- 4-14    Myers, G., "*Composite/Structured Design*," Van Nostrand Reinhold Co., 1978.  
       This is an excellent book on structured design.
- 4-15    Orr, K., "*Structured Systems Development*," Prentice Hall, 1977.  
       This book gives a better description of the Warnier-Orr method.
- 4-16    Page-Jones, M., "*The Practical Guide to Structured Systems Design*," Prentice Hall, 1988 (2nd edition).  
       An excellent introduction to structured design. The second edition ties essential analysis to structured design.

- 4-17 Rumbaugh, J., and Michael Blaha, William Premerlani, Frederick Eddy, William Lorensen, "*Object-Oriented Modeling and Design*," Prentice Hall, c1991.  
This book describes the widely use OMT methodology.
- 4-18 Ward P., and S. Mellor, "*Structured Development for Real-Time Systems*," Volume 1-3, Englewood Cliffs, N.J., Prentice-Hall, 1985.  
These books describe a methodology for specifying requirements for real-time systems. This method is an extension of Structured Analysis, which was developed by Tom DeMarco.
- 4-19 Yourdon, E., and L. Constantine, "*Structured Design: Fundamentals of a Discipline of Computer Program and Systems Design*," Prentice-Hall, Englewood Cliffs, N.J., 1975.  
This book covers a methodology on structured design which was a very popular method for designing software systems during the 1970s.

## 5 CASE Tool Economics

The papers in this section deal with the economics of using CASE tools. In particular, they address how software costs can be estimated and the economic benefits of using CASE tools.

- 5-1 Boehm, B., "*Software Engineering Economics*," Prentice-Hall, 1981.  
This landmark textbook provides a very thorough treatment of the factors that impact the cost of software development. Many examples are included as well as detailed procedures for deriving cost estimates using the CONstructive COst MODEL (COCOMO).
- 5-2 Card, D., F. McGarry, and G. Page, "Evaluating Software Engineering Technologies," *IEEE Transactions on Software Engineering*, Vol. SE-13, No. 7, July 1987.  
This paper reports on the results of a study undertaken at the NASA Goddard Software Engineering Laboratory to empirically measure software development practices, tools, and techniques to evaluate the effects of these technologies on productivity and reliability. The study concluded that there was an approximate 30% increase in reliability but no direct effect on productivity was found.

## **Appendix F: Glossary of Terms**

The following list of definitions was selected from the Glossary of Software Engineering Terminology that was an update to ANSI/IEEE Std 729-1983. The glossary was prepared by the Computer Dictionary Working Group chaired by Jane Radatz. This glossary subset relates to functionality and attributes dealing with Upper Case tools.

**Abstract Data Type.** A data type for which only the properties of the data and the operations to be performed on the data are specified, without concern for how the data will be represented or how the operations will be implemented.

**Abstraction.** (1) A view of an object that focuses on the information relevant to a particular purpose and ignores the remainder of the information. Also see: **Data Abstraction.** (2) The process of formulating a view as in (1).

**Address.** (1) A number, character, or group of characters that identifies a given device or storage location. (2) To refer to a device or storage location by an identifying number, character, or group of characters.

**Algorithm.** (1) A finite set of well-defined rules for the solution of a problem in a finite number of steps; for example, a complete specification of a sequence of arithmetic operations for evaluating sine x to a given precision. (2) Any sequence of operations for performing a specific task.

**Application Generator.** A code generator that produces programs to solve one or more problems in a particular application area; for example, a payroll generator.

**Application Software.** Software designed to fulfill specific needs of a user; for example, software for navigation, payroll, or process control. Contrast with: **Support Software; System Software.**

**Architectural Design.** (1) The process of defining a collection of hardware and software components and their interfaces to establish the framework for the development of a computer system. Also see: **Functional Design.** (2) The result of the process in (1).

**Atomic Type.** A data type each of which its members consists of a single, nondecomposable data item. Contrast with: **Composite Type.**

**Attribute.** A characteristic of an item; for example, the item's color, size, or type. Also see: **Quality Attribute.**

**Automated Verification System.** (1) A software tool that accepts as input a computer program and a representation of its specification and produces, possibly with human help, a proof or disproof of the correctness of the program. (2) Any software tool that automates part or all of the verification process.

**Background.** In job scheduling, the computing environment in which low-priority processes or those that do not require user interaction are executed. Contrast with: **Foreground.** Also see background processing.

**Background Processing.** The execution of a low-priority process while higher-priority processes are not using computer resources, or the execution of processes that do not require user interaction. Contrast with: **Foreground Processing.**

**Backup.** (1) A system, component, file, procedure, or person available to replace or help restore a primary item in the event of a failure or externally-caused disaster. (2) To create or designate a system, component, file, procedure, or person as in (1).

**Benchmark.** (1) A standard against which measurements or comparisons can be made. (2) A procedure, problem, or test that can be used to compare systems or components to each other or to a standard as in (1). (3) A recovery file.

**Black Box.** (1) A system or component of which its inputs, outputs, and general function are known but its contents or implementation are unknown or irrelevant. Contrast with:

**Glass Box.** (2) Pertaining to an approach that treats a system or component as in (1). Also see: **Encapsulation.**

**Block Diagram.** A diagram of a system, computer, or device in which the principal parts are represented by suitably annotated geometrical figures to show both the functions of the parts and their functional relationships. Also see: **Box Diagram; Bubblechart; Flowchart; Graph; Input-Process-Output Chart; Structure Chart.**

**Bottom-Up.** Pertaining to an activity that starts with the lowest level components of a hierarchy and proceeds through progressively higher levels; for example, bottom-up design; bottom-up testing. Contrast with: **Top-Down.** Also see: **Critical Piece First.**

**Box Diagram.** A control flow diagram that consists of a rectangle that is subdivided to show sequential steps, if-then-else conditions, repetition, and case conditions.

**Bubble Chart.** A data flow, data structure, or other diagram in which entities are depicted with circles (bubbles) and relationships are represented by links drawn between the circles.

**Call Graph.** A diagram that identifies the modules in a system or computer program and shows which modules call one another. Note: The result is not necessarily the same as that shown in a structure chart.

**CASE.** Acronym for Computer-Aided Software Engineering.

**Code Generator.** (1) A routine, often part of a compiler, that transforms a computer program from some intermediate level of representation (often the output of a root compiler or parser) into a form that is closer to the language of the machine on which the program will execute. (2) A software tool that accepts as input the requirements or design for a computer program and produces source code that implements the requirements or design.

**Composite Type.** A data type each of which its members is composed of multiple data items. For example, a data type called PAIRS of which its members are ordered pairs (x,y). Contrast with: **Atomic Type.**

**Computer-Aided Software Engineering (CASE).** The use of computers to aid in the software engineering process. May include the application of software tools to software design, requirements tracing, code production, testing, document generation, and other software engineering activities.

**Computer System.** A system containing one or more computers and associated software.

**Concept Phase.** (1) (ANSI/IEEE Std 1002-1987) The period of time in the software development cycle during which the user needs are described and evaluated through documentation (for example, statement of needs, advance planning report, project initiation



memo, feasibility studies, system definition, documentation, regulations, procedures, or policies relevant to the project). (2) (ANSI/IEEE Std 1012-1987) The initial phase of a software development project in which the user needs are described and evaluated through documentation (for example, statement of needs, advance planning report, project initiation memo, feasibility studies, system definition, documentation, regulations, procedures, or policies relevant to the project).

**Control Flow.** The sequence in which operations are performed during the execution of a computer program. Contrast with: **Data Flow.**

**Control Flow Diagram.** A diagram that depicts the set of all possible sequences in which operations may be performed during the execution of a system or program. Types include box diagram, flowchart, input-process-output chart, state diagram. Contrast with: **Data Flow Diagram.**

**Critical Piece First.** A system development approach in which the most critical aspects of a system are implemented first. The critical piece may be defined in terms of services provided, degree of risk, difficulty, or other criteria.

**Data.** (1) A representation of facts, concepts, or instructions in a manner suitable for communication, interpretation, or processing by humans or by automatic means. (2) Sometimes used as a synonym for documentation.

**Data Abstraction.** (1) The process of extracting the essential characteristics of data by defining data types and their associated functional characteristics and disregarding representation details. (2) The result of the process in (1).

**Data Flow.** The sequence in which data transfer, use, and transformation are performed during the execution of a computer program. Contrast with: **Control Flow.**

**Data Flow Diagram (DFD).** A diagram that depicts data sources, data sinks, data storage, and processes performed on data as nodes, and logical flow of data as links between the nodes.

**Data Structure.** A physical or logical relationship among data elements, designed to support specific data manipulation functions. Note: IEEE Std 610.5 defines specific data structures.

**Data Structure-Centered Design.** A software design technique in which the architecture of a system is derived from analysis of the structure of the data sets with which the system must deal.

**Data Structure Diagram.** A diagram that depicts a set of data elements, their attributes, and the logical relationship among them.

**Data Type.** A class of data, characterized by the members of the class and the operations that can be applied to them. For example, character type, enumeration type, integer type, logical type, and real type.

**Database.** A collection of interrelated data stored together in one or more computerized files. Note: IEEE Std 610.5 defines terminology pertaining to databases.

**Demodularization.** In software design, the process of combining related software modules, usually to optimize system performance.

**Design.** (1) The process of defining the architecture, components, interfaces, and other characteristics of a system or component. (2) The result of the process in (1).

**Design Description.** A document that describes the design of a system or component. Typical contents include system or component architecture, control logic, data structures, input/output formats, interface descriptions, and algorithms.

**Design Element.** (ANSI/IEEE Std 990-1987) A basic component or building block in a design.

**Design Entity.** (ANSI/IEEE Std 1016-1987) An element (component) of a design that is structurally and functionally distinct from other elements and that is separately named and referenced.

**Design Level.** (ANSI/IEEE Std 829-1983) The design decomposition of the software item (for example, system, subsystem, program, or module).

**Design Phase.** The period of time in the software lifecycle during which the designs for architecture, software components, interfaces, and data are created, documented, and verified to satisfy requirements.

**Design Requirement.** A requirement that specifies or constrains the design of a system or system component.

**Detailed Design.** (1) The process of refining and expanding the preliminary design of a system or component to the extent that the design is sufficiently complete to be implemented. (2) The result of the process in (1).

**Directed Graph.** A graph in which direction is implied in the internode connections.

**Dynamic Analysis.** The process of evaluating a system or component based on its behavior during execution.

**Efferent.** Pertains to a flow of data or control from a superordinate module to a subordinate module in a software system.

**Embedded Computer System.** A computer system that is part of a larger system and performs some of the requirements of that system; for example, a computer system used in an aircraft or rapid transit system.

**Embedded Software.** Software that is part of a larger system and performs some of the requirements of that system; for example, software used in an aircraft or rapid transit system.

**Encapsulation.** A software development technique that consists of isolating a system function or a set of data and operations on those data within a module and providing precise specifications for the module.

**Entity-Relationship (E-R) Diagram.** A diagram that depicts a set of real-world entities and the logical relationships among them.

**Extendability.** The ease with which a system or component can be modified to increase its storage or functional capacity.

**Feasibility.** The degree to which the requirements, design, or plans for a system or component can be implemented under existing constraints.

**Finite State Machine.** A computational model that consists of a finite number of states and transitions between those states, possibly with accompanying actions.

**Function.** (1) A defined objective or characteristic action of a system or component. For example, a system may have inventory control as its primary function. (2) A software module that performs a specific action, is invoked by the appearance of its name in an expression, may receive input values, and returns a single value.

**Hierarchical Decomposition.** A type of modular decomposition in which a system is broken down into a hierarchy of components through a series of top-down refinements.

**Implementation Requirement.** A requirement that specifies or constrains the coding or construction of a system or system component.

**Information Hiding.** A software development technique in which each module's interfaces reveal as little as possible about the module's inner workings and other modules are prevented from using information about the module that is not in the module's interface specification.

**Input-Process-Output (IPO) Chart.** A diagram of a software system or module that consists of a rectangle on the left listing inputs, a rectangle in the center listing processing steps, a rectangle on the right listing outputs, and arrows connecting inputs to processing steps and processing steps to outputs.

**Interoperability.** The ability of two or more systems or components to exchange information and to use the information that has been exchanged.

**Microarchitecture.** The microword definition, data flow, timing constraints, and precedence constraints that characterize a given microprogrammed computer.

**Modular Decomposition.** The process of breaking a system into components to facilitate design and development; an element of modular programming.

**Modularity.** The degree to which a system or computer program is composed of discrete components such that a change to one component has minimal impact on other components.

**Node.** (1) In a diagram, a point, circle, or other geometric figure used to represent a state, event, or other item of interest. (2) Note: The meaning of this term in the context of computer networks is covered in IEEE Std 610.5.

**Object-Oriented Design.** A software development technique in which a system or component is expressed in terms of objects and connections between those objects.

**Partitioning.** (ANSI/IEEE Std 830-1984) Decomposition; the separation of the whole into its parts.

**Performance Requirement.** A requirement that imposes conditions on a functional requirement; for example, a requirement that specifies the speed, accuracy, or memory usage with which a given function must be performed.

**Petri Net.** An abstract, formal model of information flow, showing static and dynamic properties of a system. A Petri net is usually represented as a graph that has two types of nodes

(called places and transitions) connected by arcs, and markings (called tokens) indicating dynamic properties.

**Portability.** The ease with which a system or component can be transferred from one hardware or software environment to another.

**Prototyping.** A hardware and software development technique in which a preliminary version of part or all of the hardware or software is developed to permit user feedback, determine feasibility, or investigate timing or other issues in support of the development process.

**Rapid Prototyping.** A type of prototyping in which emphasis is placed on developing prototypes early in the development process to permit early feedback and analysis in support of the development process.

**Real-Time.** Pertains to a system or mode of operation in which computation is performed during the actual time that an external process occurs, so the computation results can be used to control, monitor, or respond in a timely manner to the external process.

**Requirements Analysis.** (1) The process of studying user needs to arrive at a definition of system, hardware, or software requirements. (2) The process of studying and refining system, hardware, or software requirements.

**Reusability.** The degree to which a software module or other work product can be used in more than one computer program or software system.

**Shell.** A computer program or routine that provides an interface between the user and a computer system or program.

**Simulation.** (1) A model that behaves or operates like a given system when provided a set of controlled inputs. (2) The process of developing or using a model as in (1).

**Sizing.** The process of estimating the amount of computer storage or the number of source lines required for a software system or component.

**Structure Chart.** A diagram that identifies modules, activities, or other entities in a system or computer program and shows how larger or more general entities break down into smaller, more specific entities.

**Structured Design.** (1) Any disciplined approach to software design that adheres to specified rules based on principles such as modularity, top-down design, and stepwise refinement of data, system structures, and processing steps. (2) The result of applying the approach in (1).

**Taxonomy.** (ANSI/IEEE Std 1002-1987) A scheme that partitions a body of knowledge and defines the relationships among the pieces. It is used to classify and understand the body of knowledge.

**Timing.** The process of estimating or measuring the amount of execution time required for a software system or component. Contrast with: **Sizing.**

**Top-Down.** Pertains to an activity that starts with the highest level component of a hierarchy and proceeds through progressively lower levels; for example, top-down design; top-down testing. Contrast with: **Bottom Up.**

**Unidirected Graph.** A graph in which no direction is implied in the internode connections. Contrast with: **Directed Graph.**

**Validation.** The process of evaluating a system or component during or at the end of the development process to determine whether it satisfies specified requirements. Contrast with: **Verification.**

**Verification.** (1) The process of evaluating a system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase. Contrast with: **Validation.**

**Verification and Validation (V&V).** The process of determining whether the requirements for a system or component are complete and correct, the products of each development phase fulfill the requirements or conditions imposed by the previous phase, and the final system or component complies with specified requirements.

**Waterfall Model.** A model of the software development process in which the constituent activities, typically a concept phase, requirements phase, design phase, implementation phase, test phase, and installation and checkout phase, are performed in that order, possibly with overlap but with little or no iteration.

## **Appendix G: Software Technology Support Center**

## **G.1 Software Technology Support Center (STSC) Overview**

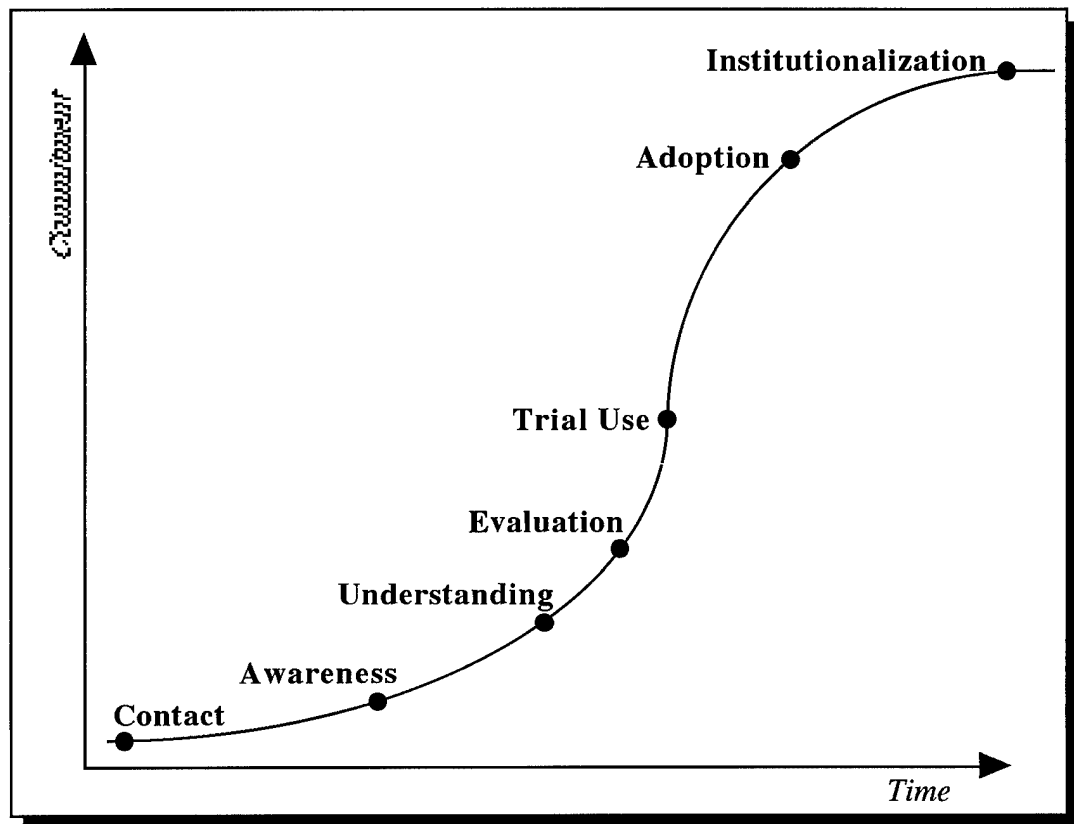
### **G.1.1 The Software Technology Support Center**

The mission of the Software Technology Support Center (STSC) is to transition technologies and exchange information to help DoD Software Development and Support Activities (SDSA) continuously improve their software quality and life cycle productivity.

A planned approach is necessary for successful transition. In general, transitioning effective practices, processes, and technologies consists of a series of activities or events that occur between the time a person encounters a new idea and the daily use of that idea. Conner and Patterson's Adoption Curve [Conner 82], shown in Figure G-1, illustrates these activities.

After encountering a new process or technology, potential customers of that technology increase their awareness of its usage, maturity, and application. If the process or technology is promising, then customers try to better understand its strengths, weaknesses, costs, and applications. These first activities in the Adoption Curve take a significant amount of time.

Next, the customer evaluates and compares the processes and technologies that show the most promise. To reduce the risk, customers usually try new processes or technologies on a limited scale through beta tests, case studies, or pilot projects. A customer then adopts processes or technologies that prove effective. Finally, refined processes and technologies become essential parts of an organization's daily process (institutionalization).



**Figure G-1. Adoption Curve**

Word processors are essential in most organization's daily operations. Yet, thirty years ago they did not exist. The institutionalization of word processors in many organizations followed a series of events similar to those identified in the Adoption Curve.

The STSC is researching and collecting information about technologies that will reduce the time and resources it takes to become aware, understand, evaluate, test, try, and adopt effective practices, processes, and technologies. The STSC has developed the following objectives to accomplish its mission:

**Technology Evaluation**

Identify, validate, classify, and evaluate effective processes and technologies.

**Information Exchange**

Facilitate the exchange of better software business practices, processes, and technologies within the DoD.

**Insertion Projects**

Analyze and improve processes, adopt new methodologies as needed, evaluate and select effective tools, receive appropriate levels of training, and perform pilot projects to try out and confirm the technology insertion efforts.



**STSC Associates**

Develop STSC Associates who can infuse effective process and technology improvements through the use of STSC products, services, and processes.

## **G.1.2 STSC Technology Transition Approach**

This section describes the STSC's approach to meeting the objectives identified in the previous section.

### **G.1.2.1 Technology Evaluation**

The first technology transition objective involves identifying, validating, and classifying processes, methods, and technologies that can potentially improve the quality or productivity of software development and maintenance. Many organizations are so focused on deadlines and customer needs that they lack the resources and time to thoroughly investigate options for improvement, leaving them vulnerable to marketing hype. The STSC has developed the infrastructure to provide information on all types of applicable technologies. Product critiques, which are essentially brief evaluations from experienced technology users, are collected. Quantitative evaluations, which are detailed, comparable, and objective, are performed on the most promising tools, methods, or processes.

### **G.1.2.2 Information Exchange**

This technology transition objective involves exposing potential customers to available technologies and, conversely, customer requirements to technology developers. Referring to the Adoption Curve, this objective focuses on contact, awareness, and understanding. STSC products that accomplish this objective include *CrossTalk* (a monthly technology report), the annual Software Technology Conference, specific technology reports, and electronic customer services.

#### **G.1.2.2.1 CrossTalk**

Over 16,000 software professionals receive *CrossTalk* monthly. This publication provides a forum for the exchange of ideas. Articles cover leading edge, state-of-the-art, and state-of-the-practice processes and technologies in software engineering.

#### **G.1.2.2.2 Software Technology Conference**

The annual Software Technology Conference is held each April in Salt Lake City, Utah. This conference brings together over 2,500 software professionals from government, industry, and academia to share technology solutions and exchange ideas and information.

#### **G.1.2.2.3 Technology Reports**

STSC technology reports provide detailed information on specific software engineering technologies, and this report is an example. The current list of reports includes:

- Software Test Technologies
- Documentation
- Project Management and Software Cost Estimation
- Requirements Analysis and Design
- Reengineering
- Process Technologies
- Software Engineering Environment
- Software Configuration Management

These reports provide awareness and understanding of each topic in preparation for evaluation and selection of corresponding technologies. Over 60,000 of these reports have been distributed. In addition to the technology reports, the following products are also available:

- Guidelines for Successful Acquisition and Management of Software Intensive Systems.
- Metrics Starter Kit and Guidelines
- Cleanroom Pamphlet

#### **G.1.2.2.4 Electronic Customer Services**

Along with the services mentioned above, the STSC also provides customers with electronic access to information via Electronic Customer Services (ECS).

#### **Software Technology Support Center STSC On-Line Services**

The Software Technology Support Center (STSC) is pleased to make available its On-Line Services to the software engineering community. We think you'll like what you find on our Bulletin Board System, World Wide Web Home Page, Lynx Browser, Gopher Client/Server, and Anonymous FTP Site.

#### **Telnet Connection**

Customers with Internet capability can connect to the STSC On-Line Services Bulletin Board System (BBS) with the command: `telnet bbs.stsc.hill.af.mil`. When connected, follow the instructions provided to bring up the STSC On-Line Services Main Menu.

### **Dial-in Connection**

Customers lacking Internet capability can connect to the STSC On-Line Services Bulletin Board via modem. Dial 801-774-6509 or DSN 775-3602. Set your device to VT emulation. Set your modem to between 2400 and 9600 bits per second, 8-bit word, no parity, and 1 stop bit. When connected, follow the instructions provided to bring up the STSC On-Line Services Main Menu.

### **World Wide Web Home Page**

The STSC Web Home Page utilizes the most recent methods for locating information on the Internet. Initially, our Web site was experimental in nature, but is now maturing into a productive and useful site. You will find an expanding number of software engineering options including hot links and a menu featuring other Web Servers. To access the STSC Home Page, connect to `http://www.stsc.hill.af.mil/` from your host terminal.

### **Lynx Browser**

The BBS Main Menu now features a Lynx Browser option. It is tailored for users desiring to explore the vast information resources of World Wide Web, but who lack full Web and graphics capability. Lynx is your avenue of access. It was developed by the University of Kansas as a text only Web browser. It brings the Web to your personal computer in ASCII text only. As a tested and proven, simple to use Web navigational tool, Lynx will be helpful to many customers. To access the STSC's Lynx Browser, Telnet to or dial into the BBS as explained above. Select Option [14] - Lynx Browser to WWW Server, on the STSC On-Line Services Main Menu.

### **Gopher Client/Server**

The STSC Gopher Client/Server allows access to the Internet for information searches and retrieval. It includes Veronica keyword search capability. You can access the STSC Gopher Client/Server through the STSC Bulletin Board System by selecting Main Menu Option [13] - Gopher Server, on the Main Menu. It can also be reached over the Internet by entering the command `gopher gopher.stsc.hill.af.mil` at your host terminal.

### **Anonymous FTP**

The STSC Anonymous FTP Site allows the transfer of files over the Internet from the STSC host to your host using the File Transfer Protocol. Some of the files in the FTP directory contain text accompanied by graphics as opposed to merely ASCII text. Special directories accommodate the graphics files. To reach the Anonymous FTP Site via Internet, enter the command `ftp ftp.stsc.hill.af.mil` at your host terminal.

### **In Conclusion...**

The Software Technology Support Center is continually striving to offer its customers new and up-to-date information in the field of software engineering. We depend on your input to expand that body of knowledge. Alert us if you know of a new Web site we should point to. If you have articles, reports, or other related documents you would like to share with the Software Engineering Community, our On-Line Services point of contact is:

George A. Klipper  
Information Manager/BBS  
OO-ALC/TISEB  
7278 4th Street  
Hill AFB UT 84056  
DSN 777-9712 (801) 777-9712 Fax (801) 777-8069  
[klipperg@software.hill.af.mil](mailto:klipperg@software.hill.af.mil)

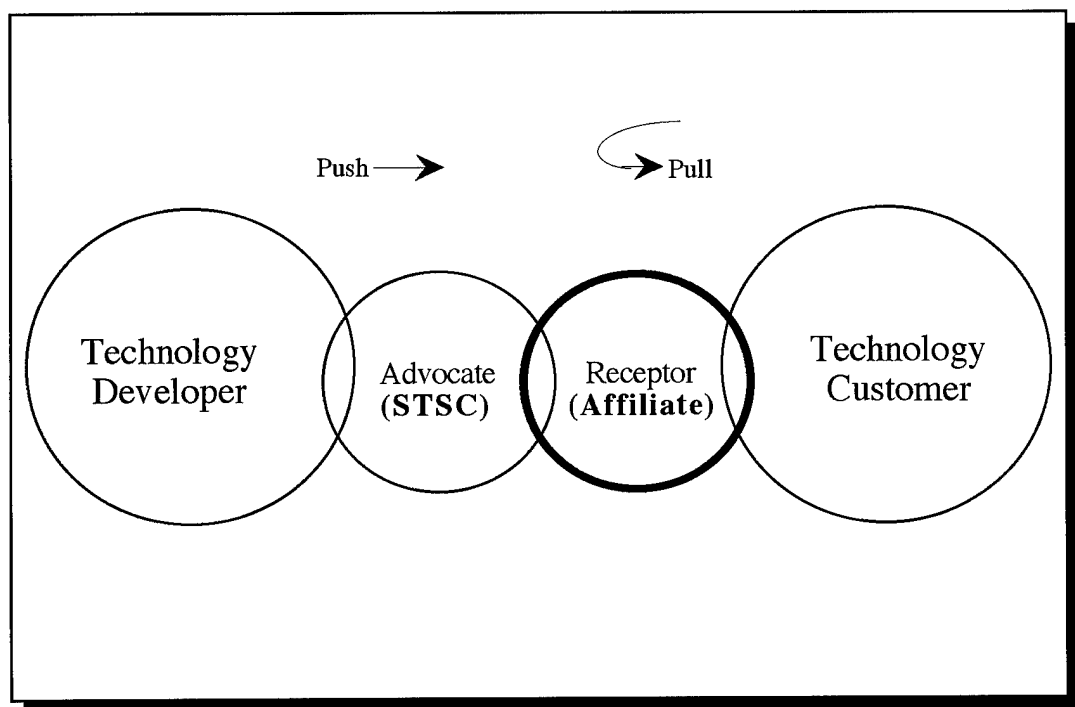
### **G.1.2.3 Technology Insertion Projects**

STSC technology insertion projects are customer oriented projects that evaluate, select, and pilot the use of new processes, methods, and technologies for a specific customer. These projects can include process definition, process improvement, methodology insertion, tool insertion, and development of a technology road map. Referring to the Adoption Curve, Figure G-1, an insertion project helps cement understanding of a process or technology, tailors an evaluation of the process or technology for the customer, and pilots the use of that process or technology with appropriate levels of training. Customers move closer to adoption of the process or technology through hands-on experience. It is important to try out technology improvements in a pilot project to confirm that the technology is appropriate for the organization and that the organization is ready and able to adopt the new technology.

#### G.1.2.4 STSC Associates

Fowler and Przybylinski [Fowler 88] propose that transitioning new technologies from a developer to a consumer requires an advocate to push the technology and a receptor to pull the technology into an organization. This concept is illustrated in Figure G-2.

Effective change comes from within the organization. The STSC Associates objective is to develop technology receptors within individual Air Force SDSAs. These receptors, STSC Associates, are trained in the use of the STSC's information, products, and services to enhance their organization's ability to incorporate advanced practices, processes, and technologies.



**Figure G-2. Transitioning Technology**

Referring to the Adoption Curve in Figure G-1, STSC Associates complete the trek to institutionalization. Associates coming from within the organization should be politically astute and aware of internal organizational requirements. They have the highest probability of influencing the adoption and daily use of effective business practices, processes, and technologies.

### **G.1.3 Embedded Computer Resources Support Improvement Program (ESIP)**

The STSC operates out of the Ogden Air Logistics Center at Hill Air Force Base, Utah, under the direction and guidance of the ESIP Program Office. An Air Force program, the ESIP has the goals of reducing the software backlog and increasing software quality and productivity. Its mission is to provide an infrastructure to assist in the transitioning of technology to support all categories of embedded computer systems throughout the acquisition cycle and improve the readiness of Air Force weapon systems. ESIP is divided into four tasks: Readiness, the STSC, Extendible Integration Support Environment (EISE), and Advanced R&D. ESIP is directed by an Air Force program management directive (PMD3118) and is led by Col. Charles Fuller. An ESIP working group has been established as a forum to share lessons learned and establish requirements for ESIP funded technology transition projects. Working group members are from the major commands, ESIP task managers, and the ESIP program office staff.